2004

MOrder Server food service

Chieh-Chou Chou

MORDER-SERVER FOOD SERVICE

A Project

Presented to the

Faculty of

California State University,

San Bernardino

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Computer Science

---

by

Chieh-Chou Chou

March 2004

MORDER-SERVER FOOD SERVICE

A Project

Presented to the

Faculty of

California State University,

San Bernardino

---

by

Chieh-Chou Chou

March 2004

Approved by:

[redacted]

Dr. Richard Botting, Chair, Computer Science

Mar 15/2004
Date

[redacted]

Dr. David A. Turner

[redacted]

Dr. Ernesto Gomez

ABSTRACT

mOrder-Server table service offers Point Of Sale functionality in a restaurant. It helps waitresses and waiters to make orders easily. Also, its management functions intuitively help the manager to manage the restaurant. The manager can easily edit the menu, check the sales report, discover the top sales, and so on. An intuitive touch screen puts the user in control with the touch of a finger, allows the manager to decrease training time and increase employee productivity.

TABLE OF CONTENTS

CHAPTER FOUR: USERS MANUAL

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

SOFTWARE REQUIREMENTS SPECIFICATION

1.1 Introduction

The content of Chapter One presents an overview of the project. The context of the problem is discussed followed by the purpose, significance of the project, and assumptions. Next, the limitations that apply to the project are reviewed. Finally, definitions of terms are presented.

1.2 Purpose of the Project

The goal of this Master's project is to develop a Point Of Sale system which helps owners to cut restaurant losses, improve customer service, reduce employee mistakes, and increase profitability. mOrder-server is touch screen operated Point Of Sale, easy to follow screen layouts, and the simplicity of Windows based interface, the user will be able to use the software in a matter of minutes. Moreover, there is no prior computer knowledge required from users so waitresses and waiters do not have to memorize product codes. It saves time and money training staff.

1.3 Context of the Problem

There are three major problems in the target restaurant. First, the restaurant uses a common cash register, a calculator with a receipt printer. Therefore, staffs have to memorize item prices and item codes. There are over 300 items in the restaurant providing the menu for my project. It takes about 2~3 weeks to train a new staff memorizing the most popular product codes. Second, the manager can not analyze the sales. The manager wants to eliminate slow-selling products but the cash register does not have statistical reports. Finally, in some cases, the staff does not charge for his/her friends, when they eat at the restaurant. It is a loss for the restaurant.

1.4 Significance of the Project

mOrder-Server is designed for touch screen operated Point Of Sale terminal. Intuitive touch screens put the staff in control with the touch of a finger. The staffs easily follow screen layouts. They do not have to memorize any product codes and prices. It can reduce employee mistakes and increase profitability. Moreover, the manager knows what items are popular. The last one and most important is that because all orders have to be inputted to the mOrder-Server system, order information will be

sent to mOrder-Kitchen[1] through a network. Cooks have to follow mOrder-Kitchen to cook. mOrder-Server can record every order. It can prevent losses like when staff do not charge his/her friends.

1.5 Assumptions

The following assumptions were made regarding the project:

1. All waitresses only serve the tables that they are responsible for.
2. All cooks only cook the orders showing in the mOrder-Kitchen which is developed by Li Qiu[1].

1.6 Limitations

For mOrder-Server, it is useful for managers to manage business. However, to be commercial, it must have good looks.

1.7 Definition of Terms

The following terms are defined as they apply to the project.

Table 1. Definition, Acronyms and Abbreviations

GUI	Graphical User Interface. The graphical representation of physical or pseudo-physical objects (such as buttons, trees, and lists) that allow the user to direct the flow of the program through the use of a mouse or other pointing device.
.NET	Microsoft® .NET is the Microsoft XML Web services platform. XML Web services allow applications to communicate and share data over the Internet, regardless of operating system, device, or programming language.
SQL	SQL (Structured Query Language) is a standard interactive and programming language for getting information from and updating a database.
Visual Studio 6.0	It is an integrated development environment (IDE) from Microsoft in which a programmer uses a graphical user interface (GUI) to choose and modify preselected sections of code written in the choice programming language.
Windows XP	A member of the Microsoft Windows family of operating systems. Windows XP was released in 2001 in two versions: Windows XP Home Edition for home use and Windows XP Professional for advanced home computing, businesses, and larger organizations. Windows XP features a new visual design that simplifies navigation and search capabilities, improved file management, additional media and Web publishing capabilities, an improved system for device discovery and installation, and advanced features for mobile computing.
Microsoft Access	A Microsoft software product is primarily a data management tool (database software). Microsoft Access has tools to enter, edit, and index data and to retrieve data via custom forms and reports. It also contains Visual Basic for applications.
SQL Server	Structured Query Language server. A computer that provides client computers with highly efficient access to database files.
ADO	ADO provides developers with a powerful, logical object model for programmatically accessing, editing, and updating data from a wide variety of data sources through OLE DB system interfaces. The most common usage of ADO is to query a table or tables in a relational database, retrieve and display the results in an application, and perhaps allow users to make and save changes to the data.
POS	Point Of Sale

1.8 Organization of the Project

The project portion is divided into five chapters. Chapter One provides software requirements specification, an introduction to the context of the problem, purpose of the project, significance of the project, limitations, and definitions of terms. Chapter Two consists of the software design. Chapter Three documents the steps used in testing the project. Chapter Four presents the user's manual from the project. Chapter Five presents conclusions drawn from the development of the project. The Appendices containing the project follows Chapter Five. Finally, the references for the project are presented.

CHAPTER TWO

SOFTWARE DESIGN

2.1 Introduction

Chapter Two consists of a discussion of the software design. It includes preliminary design, detailed design, and software design.

2.2 Preliminary Design

Before doing the project, I considered which development tools and databases apply to the project. C++, comparing with Visual Basic, is more complex and takes more time to achieve the same goal. Also, C++ is hard to debug. The benefits of Visual Basic are that it is easy to develop an application and maintain the code. Therefore, I choose Visual Basic as the development tool. For database, I consider Microsoft SQL server and Microsoft Access. SQL server performs rapid, sophisticated analysis on large and complex data sets using multi-dimensional storage. The orders of the restaurant are not over 500 per day. Also, there are no more than 4 computers in the restaurant. Microsoft Access is enough to handle such data and much cheaper than SQL server. Therefore, I choose Microsoft Access as the database in my project.

2.3 Detailed Design

There are two major parts of the mOrder-Server. One is the order function and another one is management. A waitress logins with his/her ID and password, places orders and then checks out. Another one is management. A manager logins as "Administrator" and s/he can add, delete, or update data (store information, employees, menus and so on) in mOrder-Server system. The order functions of a manager is the same as a waitress. Figure 1 is the state diagram of order function.

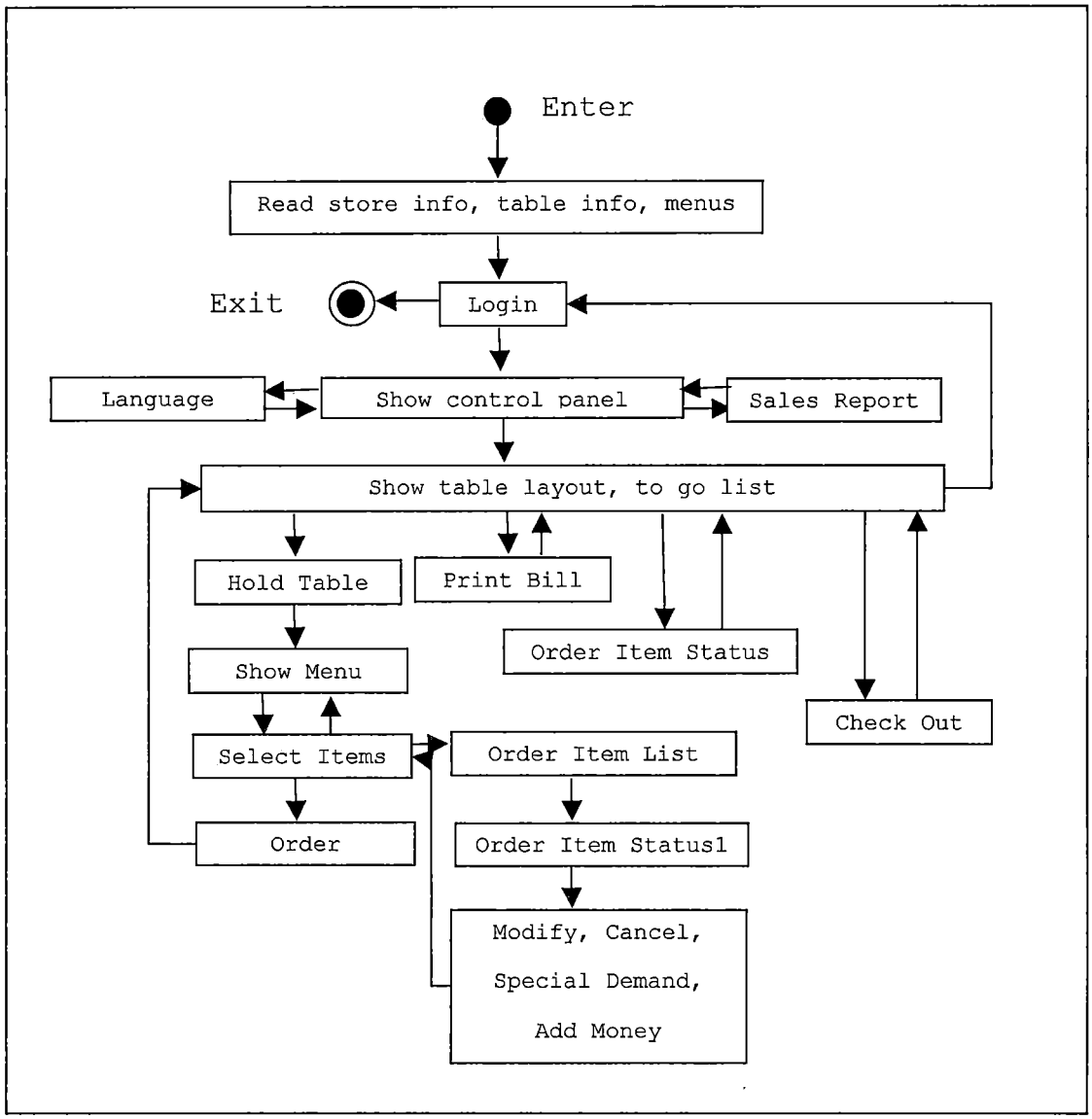


Figure 1. Waitress State Diagram

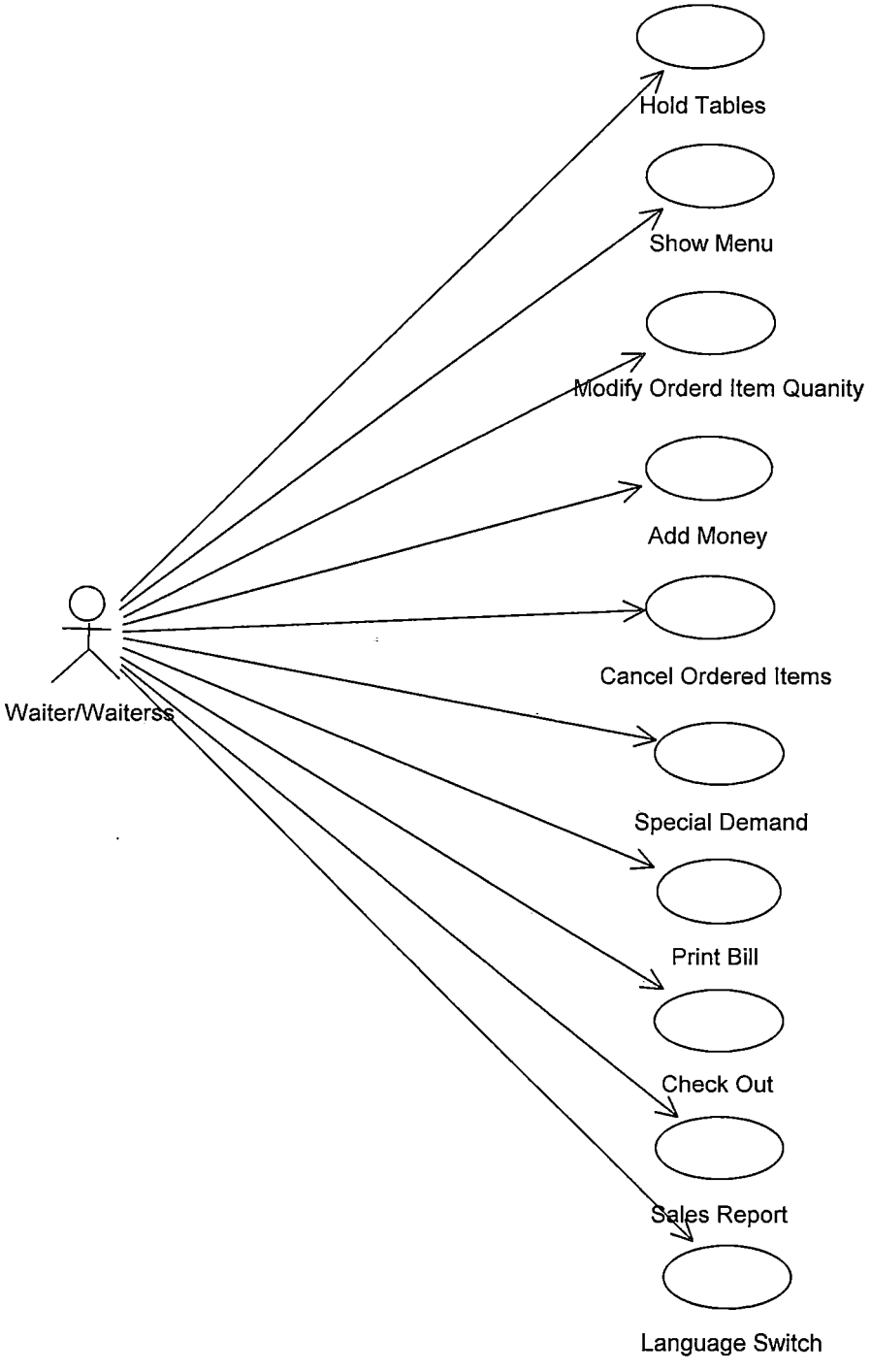


Figure 2. Waitress Use Case Diagram

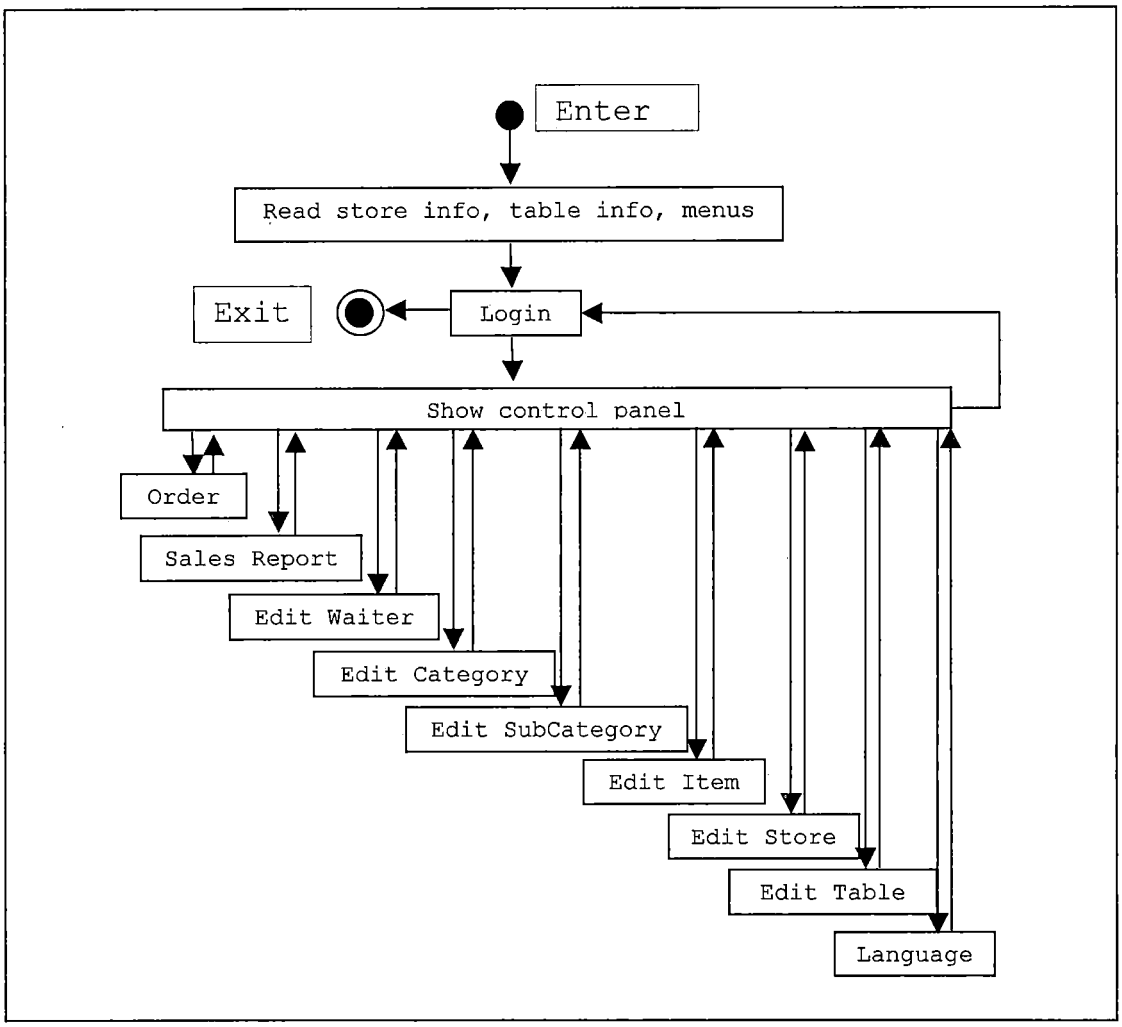


Figure 3. Manager State Diagram

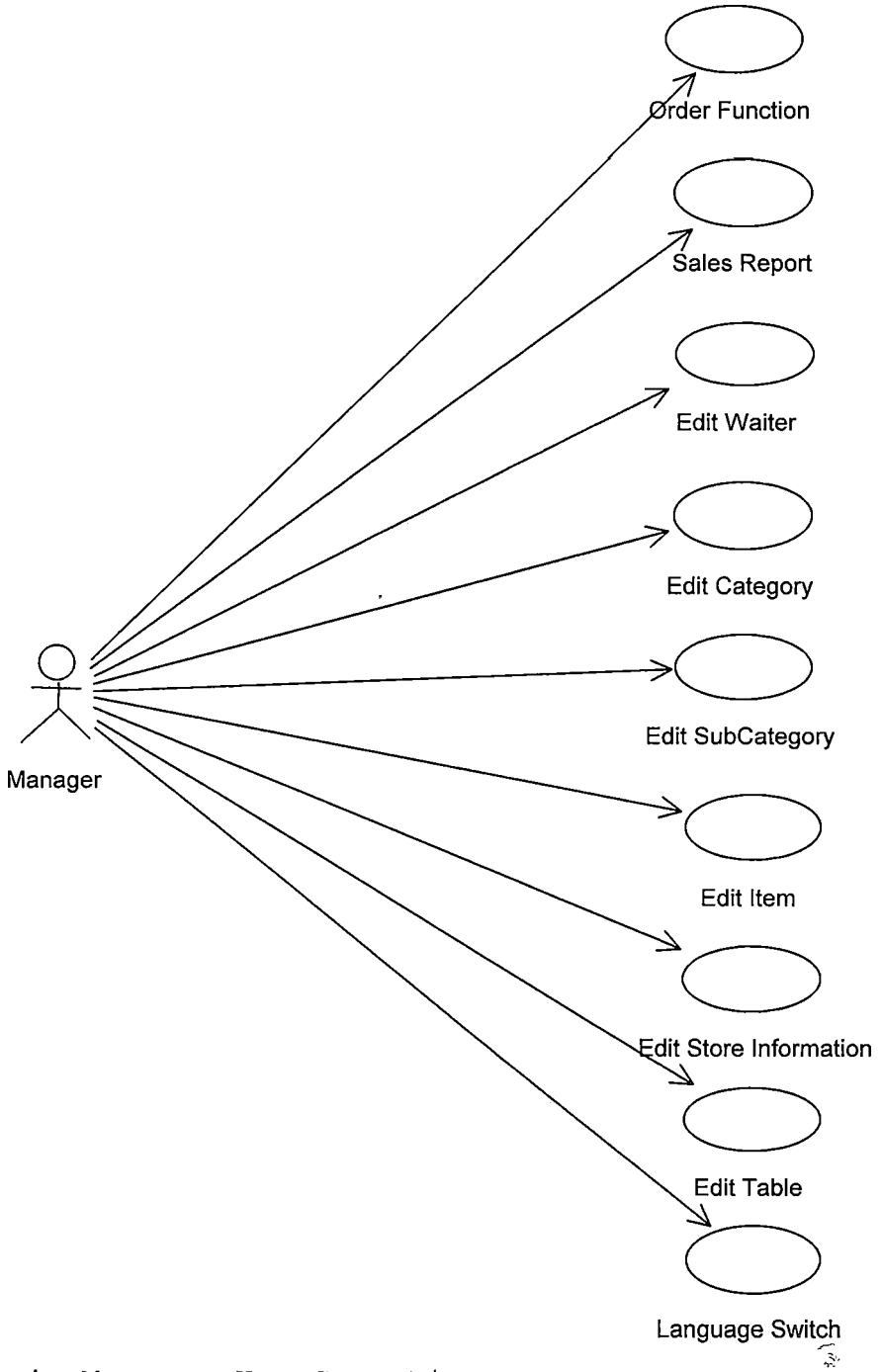


Figure 4. Manager Use Case Diagram

2.4 Software Design

Software design of mOrder-Server system is divided into three parts. They are database design, waitress order function and manager management.

2.4.1 Database Design

For mOrder-Server database application, the restaurant database that keeps track of order items, ordered item status. It stores employee information, store information, category, subcategory and item information. The ER diagram is shown in Figure 5.

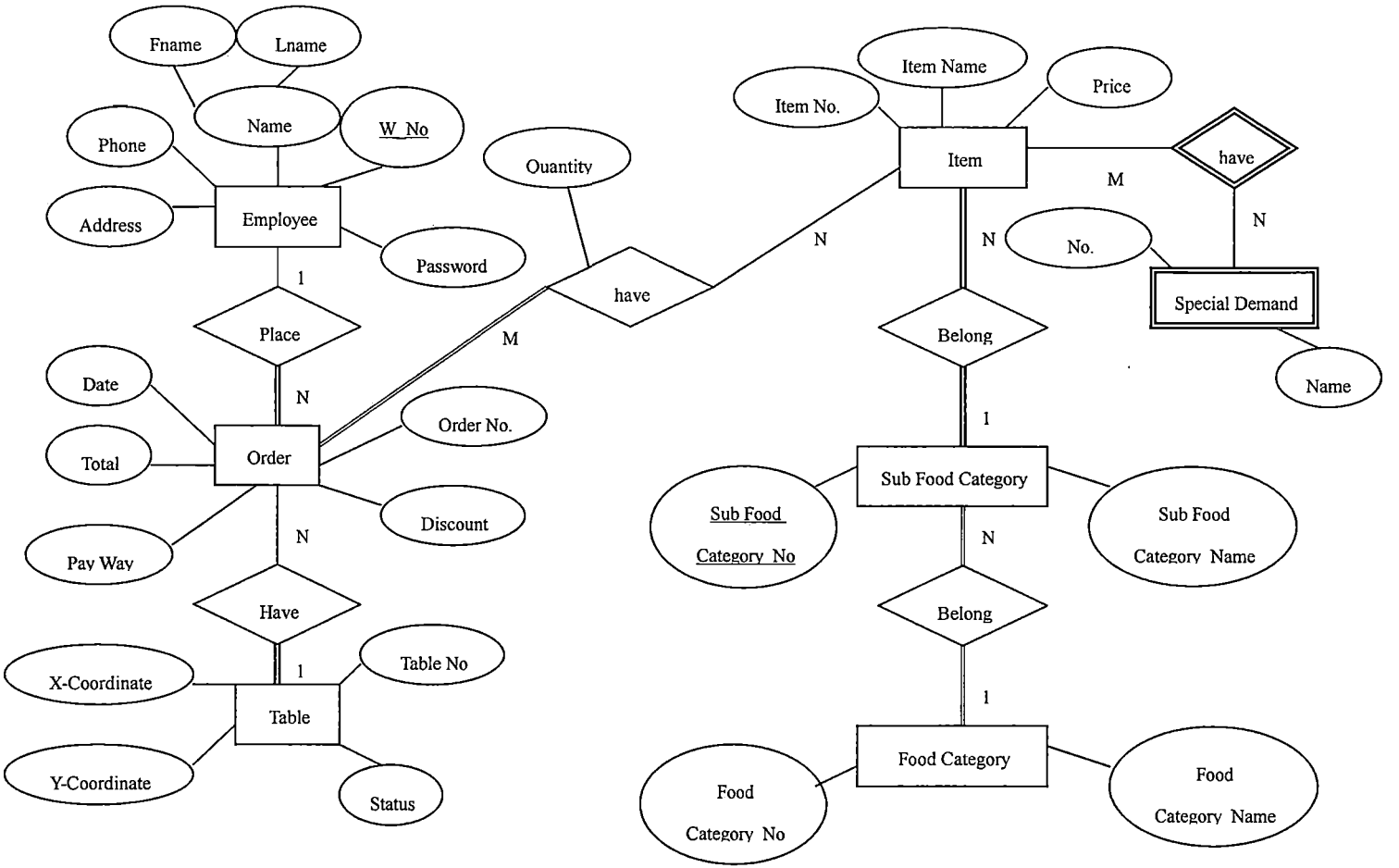


Figure 5. mOrder-Server ER Diagram

For each employee, the database maintains password, name, address and phone number. A waitress can place many orders, but an order can be only placed by one waitress. So, the relationship between an order and waitress is N: 1. The participation of order is total, whereas that of employee is partial.

An order has the attributes order number, total, pay way, date, and discount. Each order is related to a table whose attributes are table number, X coordinate, Y coordinate, and available status. An order is for one table, but a table can have many orders. The relation between table and order is 1:M. The participation of table and order are total.

Each order has at least one item and an item can be in many orders. So, the relationship between order and item is M: N. Both participations are total. An item has the attributes, item number, item name, cooking sign (show the item which is cooking or not), and price. Each item belongs to a sub category and each sub category belongs to a category.

The relationship between item and sub category is 1:1. Both participations are total. The relationship between sub category and category is 1:1. Both participations are total. In the order items table, order number and item

number is a composite key. It is more complex if a customer orders the same item more than once. For example, a customer orders a cake first time and he wants to order again. The system has to check if the order number and item number exist. If yes, mOrder-Server updates the quantity of the database. It makes the program more complex and easy to make mistakes. I create a primary key, called serial number (automatically generated by database), replacing composite key (order number and item number). Therefore, mOrder-Server does not have to check if the order number and item number exist or not. The job only inserts a new order number, item number and quantity into the database. See figure 6. the relationship diagram.

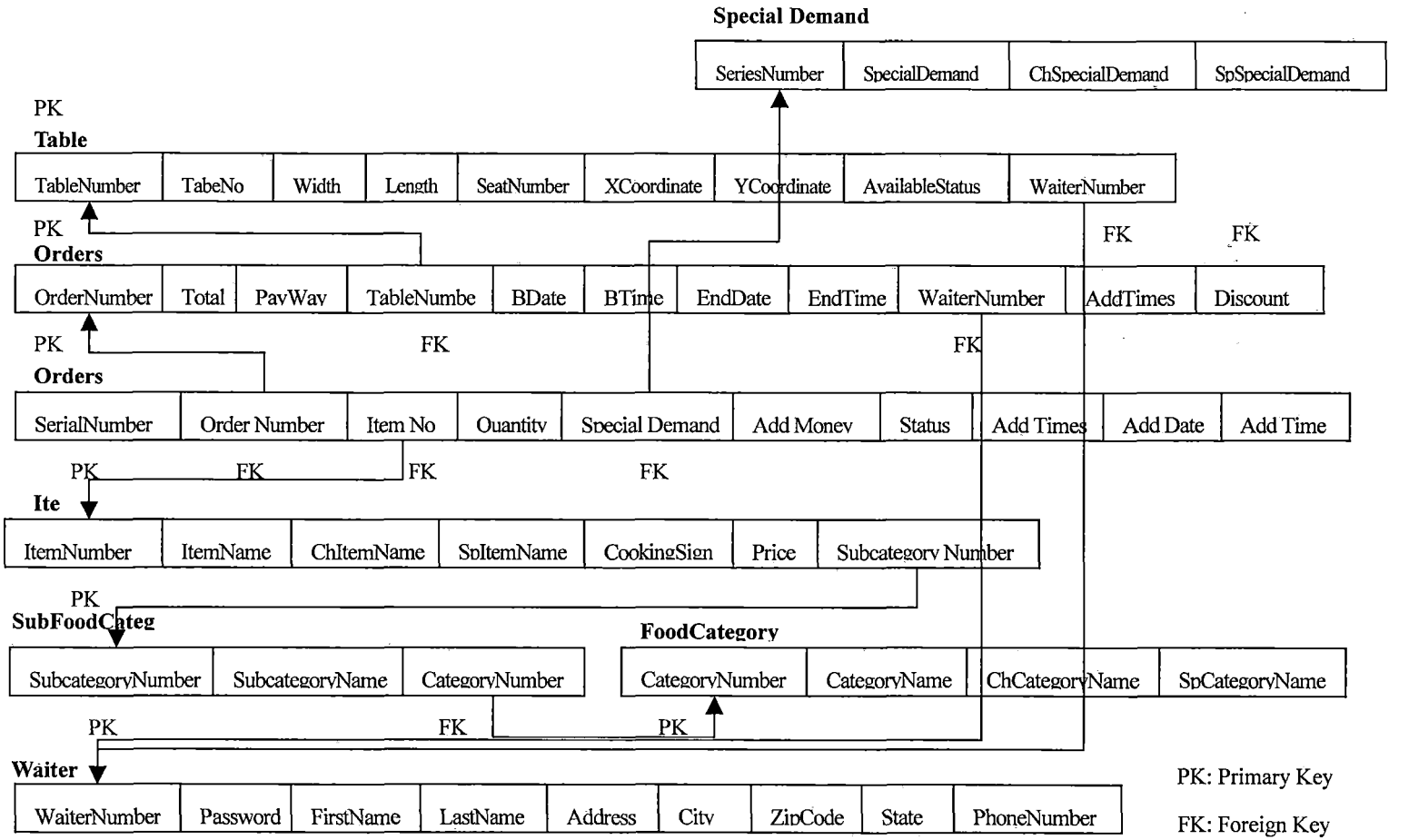


Figure 6. Relationship Diagram

2.4.2 Waitress Function

mOrder-Server has two functions for a waitress, order function and sales report. First, mOrder-Server reads table information and menus. The SQL is as follows:

```
sTableQuery = "SELECT TableNumber, TableNo, AvailableStatus, SeatNumber,
XCoordinate, YCoordinate, Width, Length, Shape FROM Tables WHERE
HallNumber=1"
```

Figure 7. Read Table Information

```
sReadFoodCategory Query = "SELECT CategoryName, ChCategoryName FROM
FOODCATEGORY"
```

Figure 8. Read Food Category

```
sReadItemInfoQuery = "SELECT ItemNumber, ItemName, Price FROM Item,
Subfoodcategory WHERE item.SubcategoryNumber =
Subfoodcategory.SubcategoryNumber AND CategoryNumber= CStr(j) ORDER BY
ItemNumber ASC"
```

Figure 9. Read Item Information

```
sReadToGoOrderInfoQuery = "SELECT OrderNumber, BeginningDate,
BeginningTime, Total, AddTimes FROM Orders WHERE EndDate IS NULL AND
TableNumber= (SELECT TableNumber FROM Tables WHERE SeatNumber='0')"
```

Figure 10. Read To Go Order Information

A waitress has to login to the system. If this passes, the control panel will show up. It has two options for a waitress. In order function, it will show table layout and to go list.

The next step, a waitress has four choices. The first one is "add item". mOrder-Server shows up menus and items. All ordered items are stored in memory before pressing "confirm" button. After pressing "confirm" button, all ordered items are inserted to the database.

```
sInsertOrderItems2DBQuery = "INSERT INTO OrderItem(OrderNumber,
ItemNumber, Quantity, SpecialDemand, CookingSpecialDemand, AddMoney,
Status, AddTimes, AddDate, AddTime) VALUES (orderNumber ,
OrderItemNumberLabel(i).Caption ,OrderItemQuantityLabel(i).Caption ,'oderItemD
emandLabel(i).Caption' ,'PrOrderItemDemandLabel(i).Caption',
OrderItemAddMoneyLabel(i).Caption ,0, CStr(addTimes), 'CStr(Date)','
CStr(Time)')"
```

Figure 11. Insert Ordered Items to Database

The second one is printing bills. The printer object in VB can not accurately get printer information. So, the output is not automatically at the correct position. I refer to the programmer's manual[5] for the codes to

position it correctly. The following table shows the Star Mode commands that are supported by the Star receipt printer.

Table 2. Control Code List

Function	Control Codes	Hexadecimal Codes
Centering	<ESC><GS>"a""1"	1B 1D 61 31
LeftJustification	<ESC><GS>"a""0"	1B 1D 61 30
FontSize(Width) FontSize(Height)	<ESC>"W""1 or 0" <ESC>"h""1 or 0"	1B 57 31 1B 68 31
Full Cut Command to the auto cutter	<ESC>"d""2"	1B 64 32
Linefeed	<LF>	0A

The major code of the printer object is as follows:

```
Public Sub OpenPrinter()
   Open "lpt1" For Output As #1
End Sub

Public Sub Center()
   Print #1, Chr$(&H1B); Chr$(&H1D); Chr$(&H61); Chr$(&H31);
End Sub

Public Sub LeftJustification()
   Print #1, Chr$(&H1B); Chr$(&H1D); Chr$(&H61); Chr$(&H30);
End Sub

Public Sub FontSize(s As Integer)
   Print #1, Chr$(&H1B); Chr$(&H57); Chr(s); Chr$(&H1B); Chr$(&H68); Chr$(s);
End Sub

Public Sub ClosePrinter()
   Print #1, Chr$(&H1B); Chr$(&H64); Chr$(&H32) 'cut function
   Close #1
End Sub
```

Figure 12. StarPrinter Object

The third one is checking order item status. In this function, a waitress can check what items have already ordered. A waitress can also cancel the items which are not yet cooked.

The last one is the payment function. In the function, the database will be updated by the following SQL:

```
sPaymentQuery = "UPDATE Orders SET Total =
FormatNumber(TotalLabel.Caption, , vbTrue, , vbFalse),PayWay ='str1',
Cashier='WaiterNumber' WHERE OrderNumber = orderNumber"
```

Figure 13. Update Order

and then update table status as available.

```
sSQLQuery = "UPDATE Tables SET AvailableStatus = str2 , WaiterNumber =str4
WHERE TableNumber = str3 AND AvailableStatus = Not (str2)"
```

Figure 14. Update Table Status

2.4.3 Management Function

A manager can access all functions that a waitress has. Besides, a manager has additional functions, editing waiter, editing category, editing sub category, editing item, editing store information, and editing table. The editing function uses ADO data control to manage the connection between mOrder-Server and the database and allows the data-aware control to view and manipulate the data in a database.

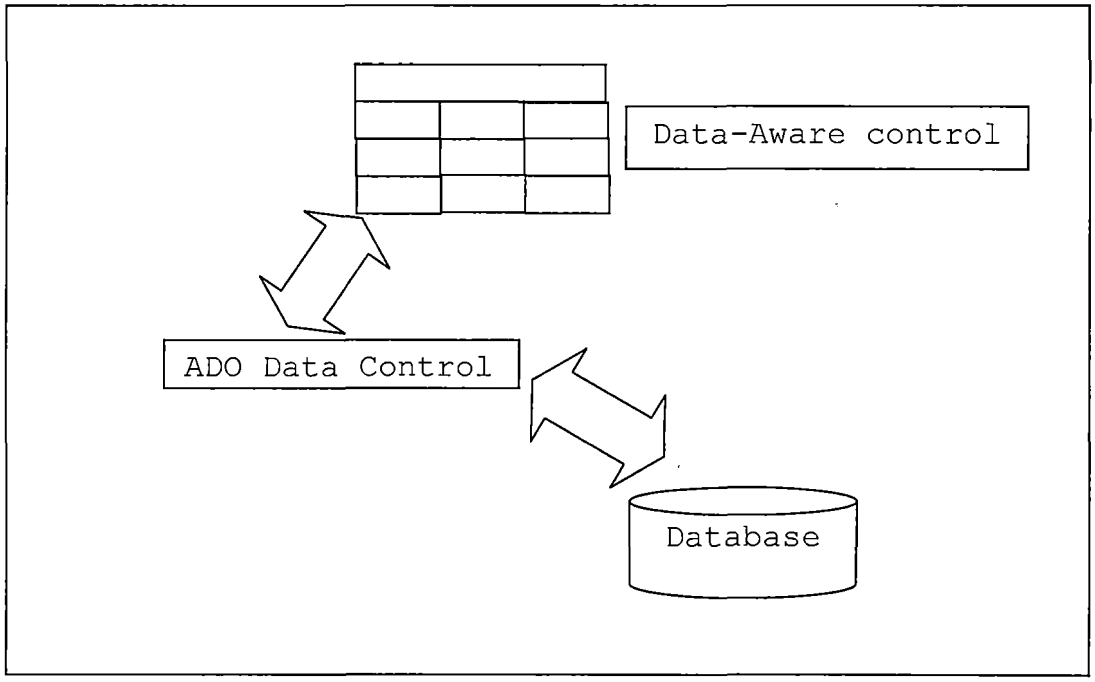


Figure 15. ADO Data Control

CHAPTER THREE

SOFTWARE QUALITY ASSURANCE

3.1 Introduction

Chapter three documents software quality assurance. mOrder-Server is a database server. In this chapter, I record checking the database and application. It includes the unit test plan and the integration test plan.

3.2 Unit Test Plan

Table 3. Database Unit Test Plan

Unit Tested	Test Performed	Results
Access database engine	Check the Microsoft Jet engine.	Pass

Table 4. Waitress Unit Test Plan

Read Table Information	Show table layout and available status	Pass
Read Category, Subcategory and Item Information	Show items by different subcategory.	Pass
Read Waitress Information	The users login to the system	Pass
Control Panel	Different users have different functions	Pass
Order Function	Place order and ordered items will show on order item list.	Pass
Cancel Order	Click ordered item list and cancel items.	Pass
Special Demand	Click ordered item list and add special demand to selected items.	Pass
Modify Quantity	Change the quantity of the ordered items.	Pass
Add Money	Add additional money to selected orders.	Pass
Print Receipt	Ordered items, subtotal, tax, and total output to the printer.	Pass
Ordered Item Status	Show ordered item status.	Pass
Payment	Show methods of payment, and numerical keyboard.	Pass
Order Confirm	Insert ordered items to database.	Pass
Check Out	Insert total to order table and update table available status as true.	Pass
Sales Report	Read daily sales information	Pass

Table 5. Manager Unit Test Plan

Order Function	The same as waitress order function	Pass
Sales Report	The same as waitress order function	Pass
Quantity Chart	Show top 5 items quantity.	Pass
Monetary Chart	Show the amount of top 5 items.	Pass
Delete orders	Delete orders from order history	Pass
Deleted order item list	Show the history of deleted items.	Pass
Edit Waitress	Show, add, delete, and edit waitress information.	Pass
Edit Category	Show, delete, and edit category information.	Pass
Edit Sub Category	Show, add, delete, and edit sub category information.	Pass
Edit Item	Show, add, delete, and edit item information.	Pass
Edit Table	Show, add, delete, and edit table information.	Pass
Edit Store Information	Show, add, delete, and edit store information.	Pass
Language Switch	Select English or Chinese	Pass

3.3 Integration Test Plan

mOrder-Server integrates with mOrder-Pocket PC[1] and mOrder-Kitchen[1]. The integration test plan tests as follows:

Table 6. Integration Test Plan

mOrder-Server with kitchen computer	Place an order in mOrder-Server, and the ordered items information will be inserted into database. The ordered items information will be sent to kitchen computer.	Pass
mOrder-Server with pocket PC	Hold a table or place an order on pocket PC, and the ordered item information will be sent to mOrder-Server. Press a refresh button in mOrder-Server, the table is hold and ordered item information shows on ordered item list.	Pass

CHAPTER FOUR

USERS MANUAL

4.1 Introduction

Included in chapter four was a presentation of the user's manual of the project. The manual includes waitress function and management function.

4.2 Waitress Function

4.2.1 Login

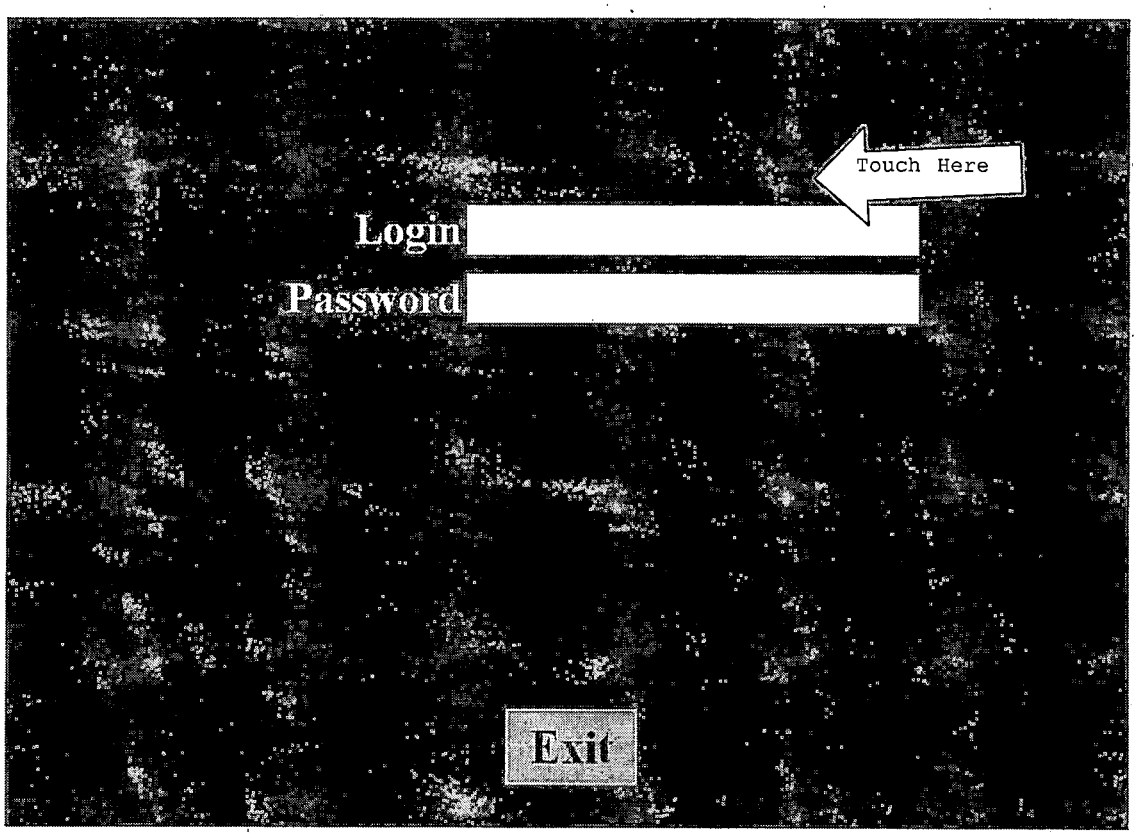


Figure 16. Login

If mOrder server operates on touch screen monitor, touch login label. If mOrder server operates on non touch monitor, click the login label. It will show a keyboard as follows:

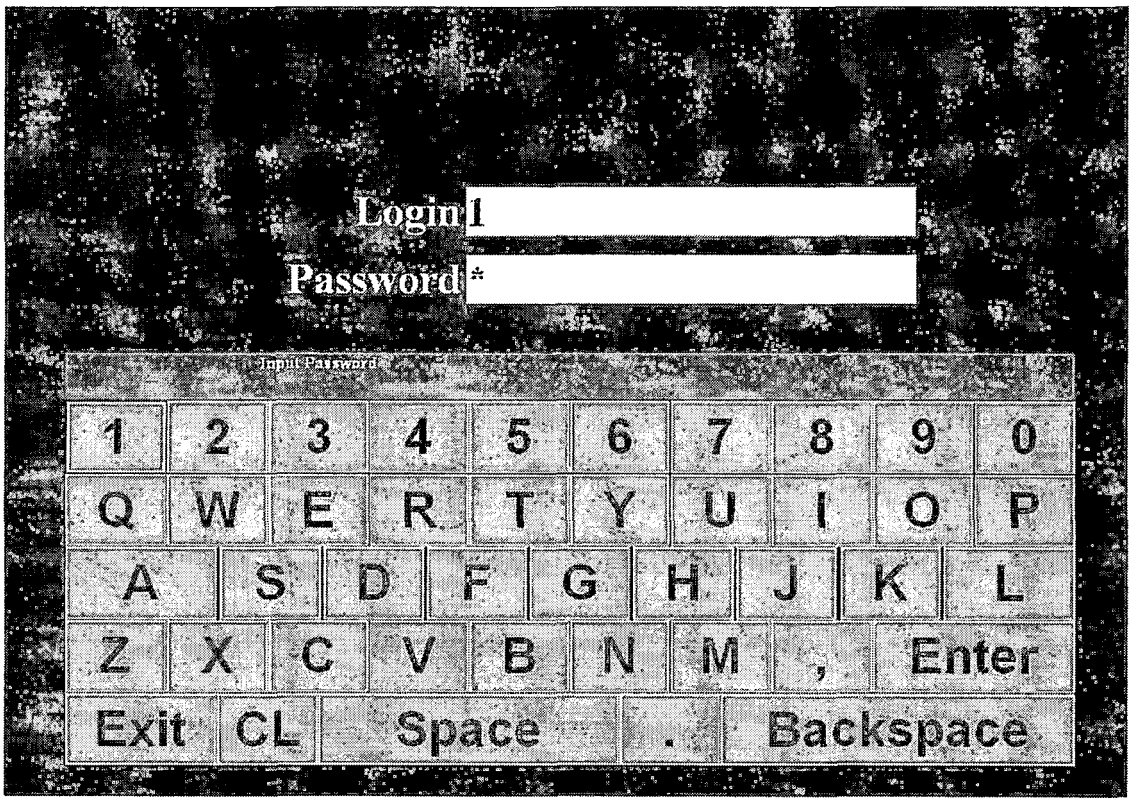


Figure 17. Login_1

Input the username and password. If username and password do not match, login and password label will be cleared. Please input again.

4.2.2 Control Panel

Figure 18. Control Panel

A waitress has two functions for mOrder server. One is order function and another is sales report.

4.2.3 Order Function

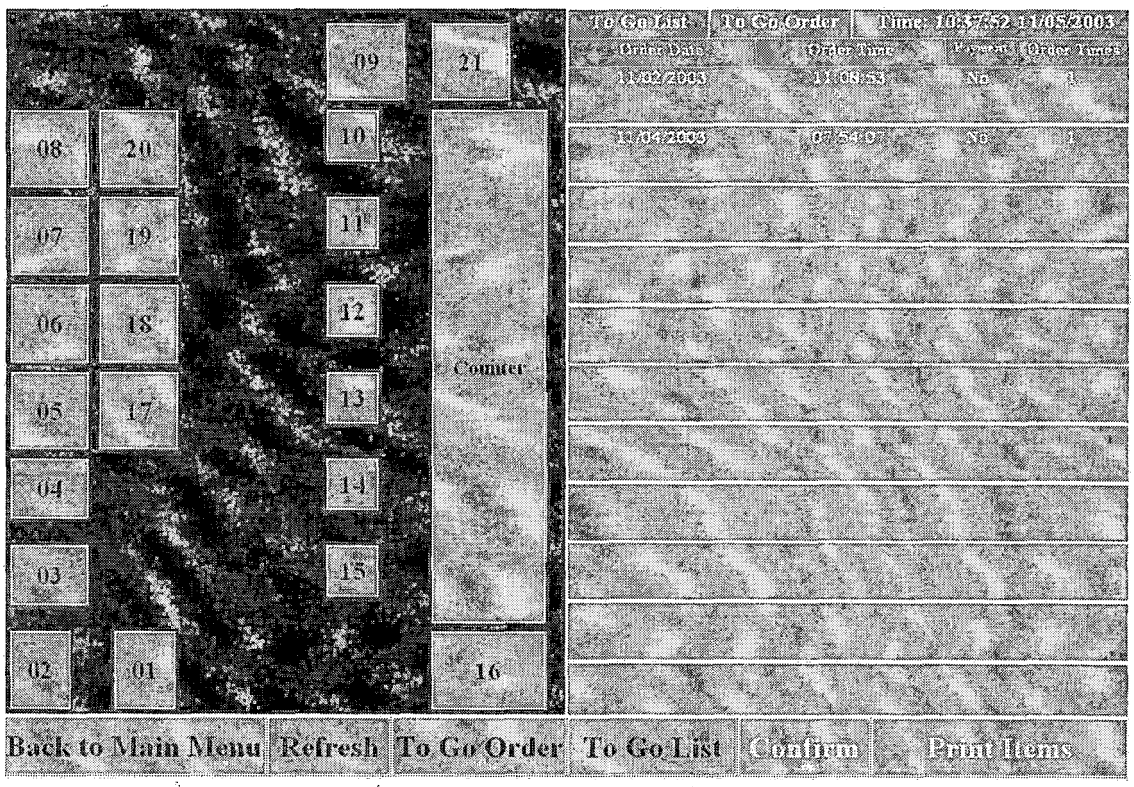


Figure 19. Order Function

The left side is the restaurant table layout and the right side is the to go order list or ordered item list.

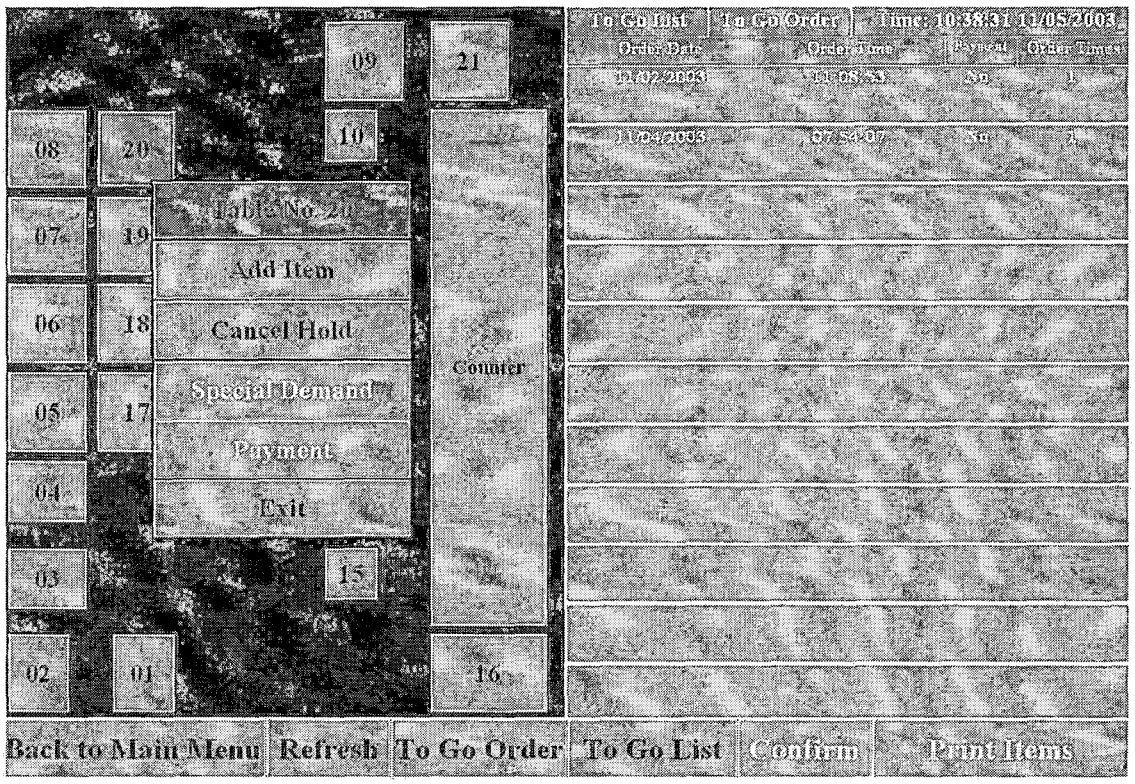


Figure 20. Order Function_1

If customers want to eat in the restaurant, a waitress clicks the table number from 01 to 21. It will pop up a menu and the color of the table will become unavailable status.

Click "Add Item", the menu will show up.

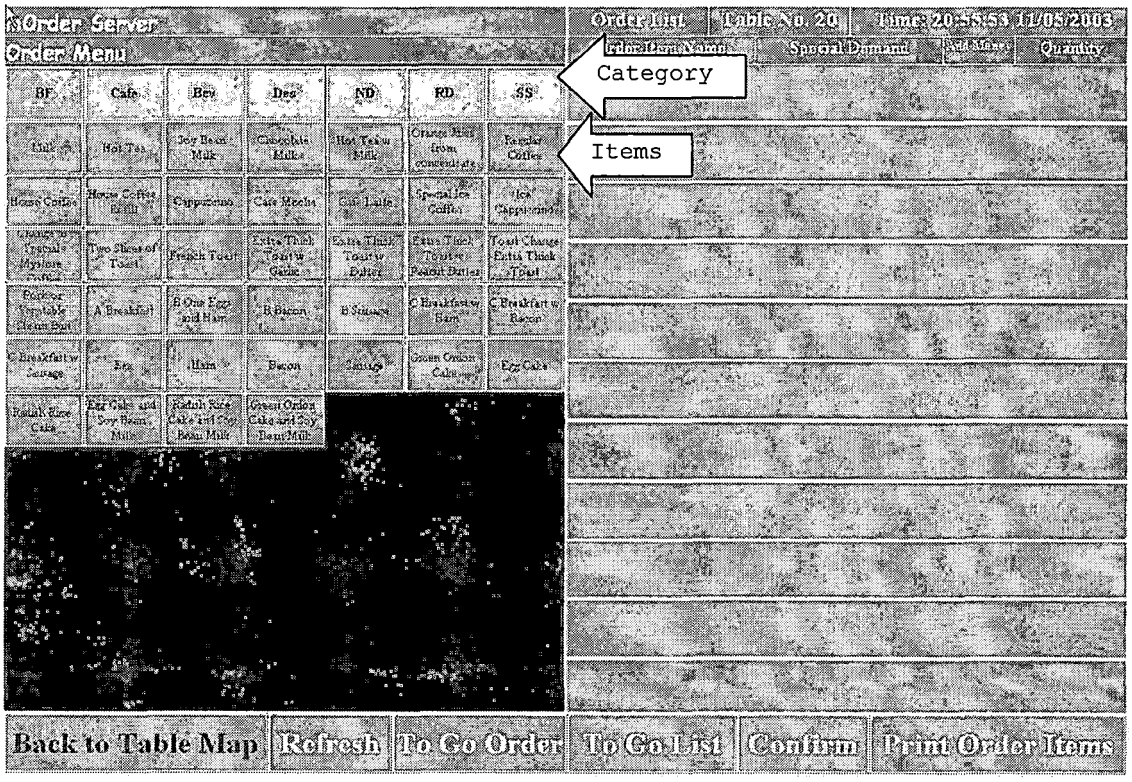


Figure 21. Menu

Click different categories and show different items.

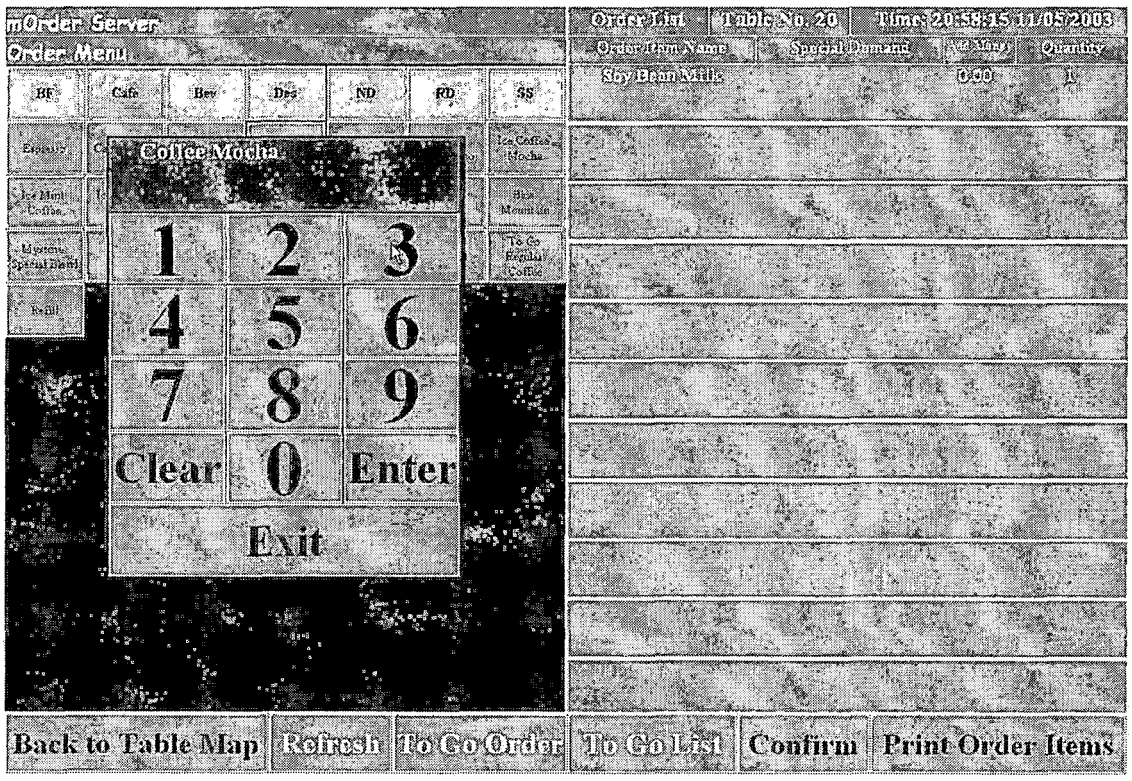


Figure 22. Place Order

Click item button, input quantity and press enter button. The ordered item will show on the order list.

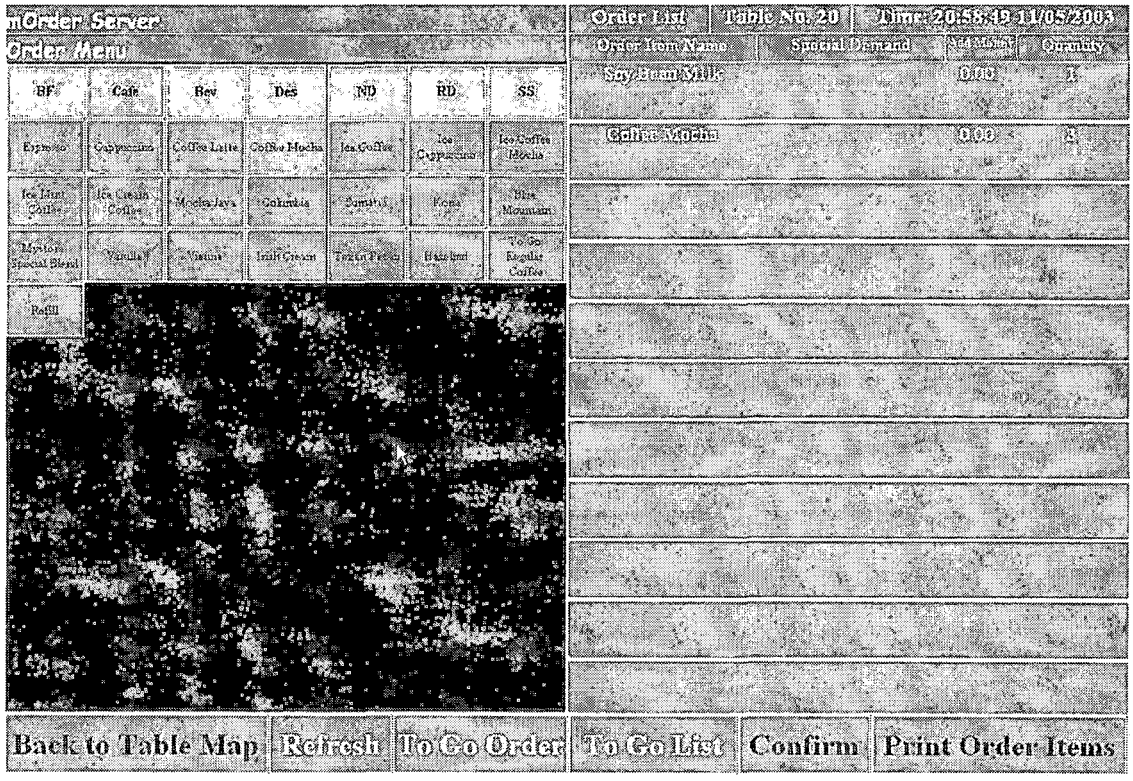


Figure 23. Order Confirmation

Click confirm button, and ordered item list will input to the database. The menu frame changes to table map frame.

4.2.4 Modify Quantity

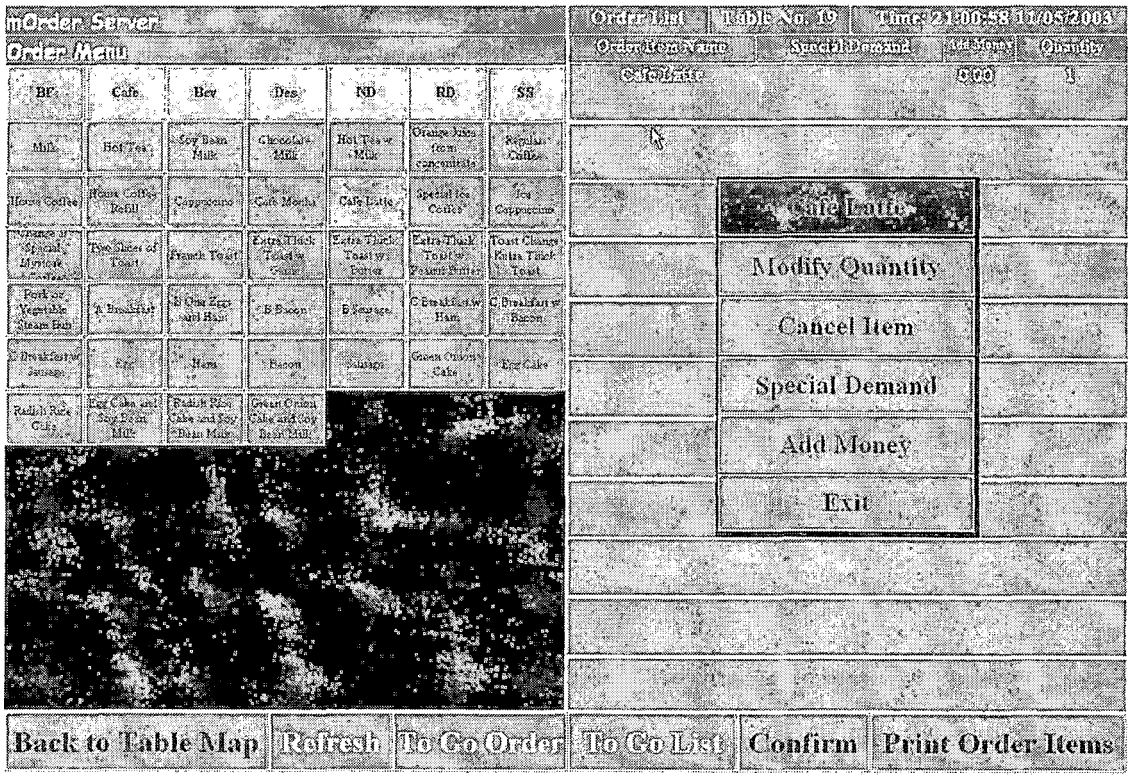


Figure 24. Modify Quantity

Click items on the order list.

Click "Modify Quantity".

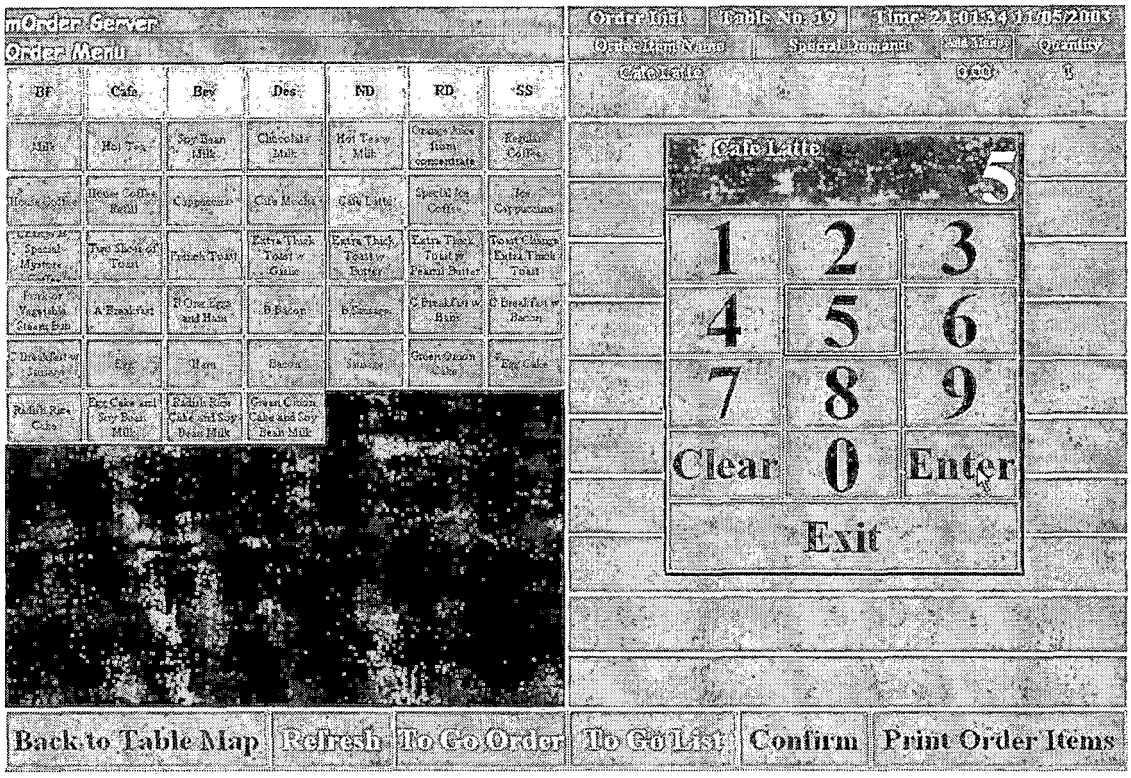


Figure 25. Modify Quantity_1

Click the number and then press enter. The quantities of the selected items will be modified.

4.2.5 Cancel Ordered Items

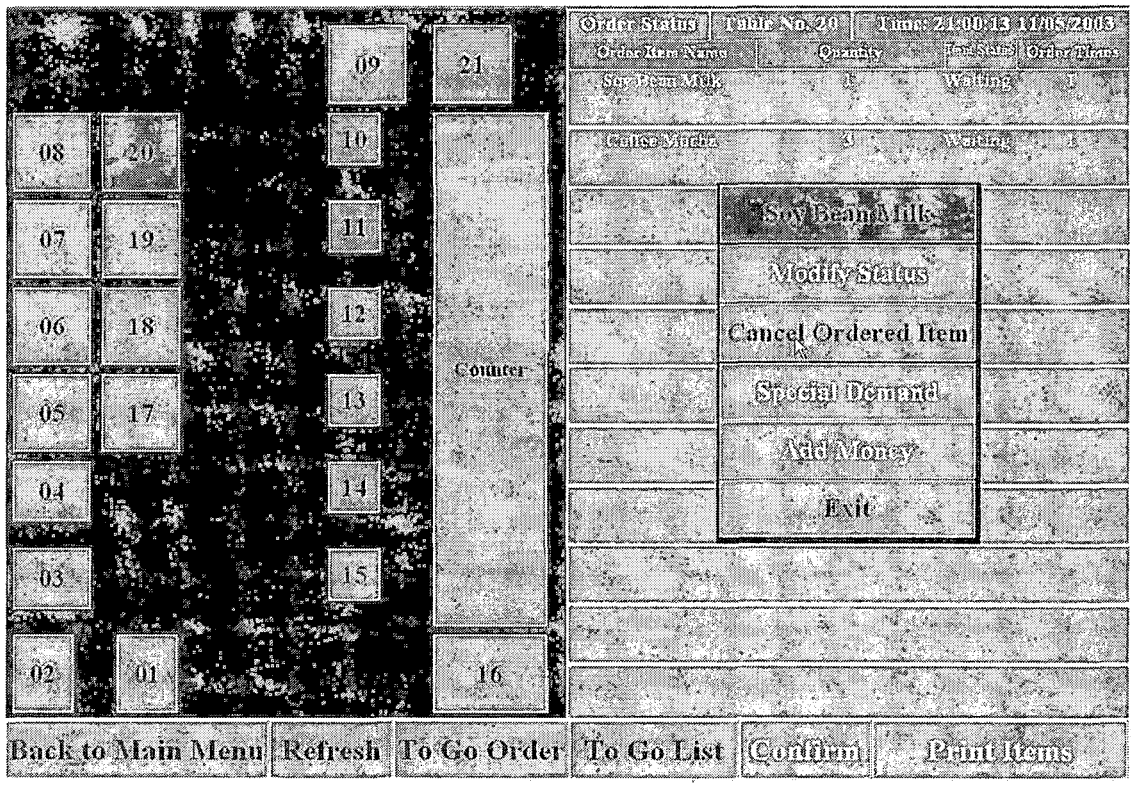


Figure 26. Cancel Ordered Items

Click ordered items, and "Cancel Ordered Item". The selected ordered item will be canceled. If the "Item Status" shows cooking, the selected item can not be canceled.

4.2.6 Special Demand

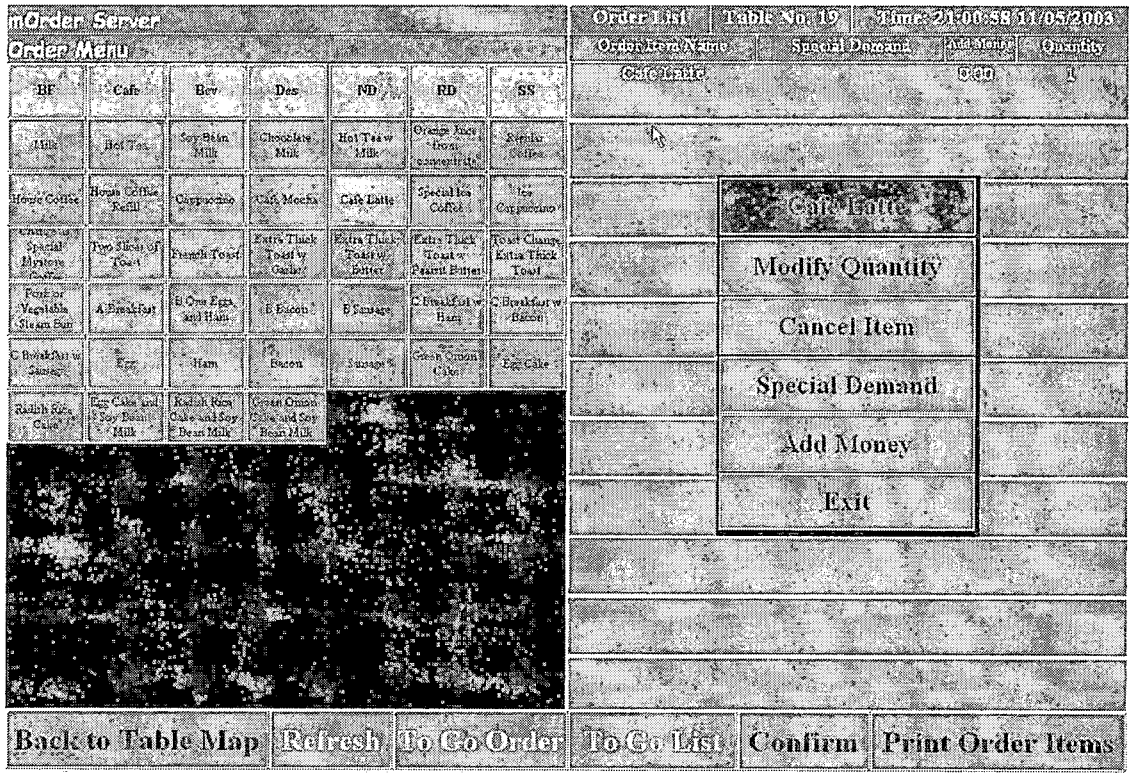


Figure 27. Special Demand

Click "Special Demand"

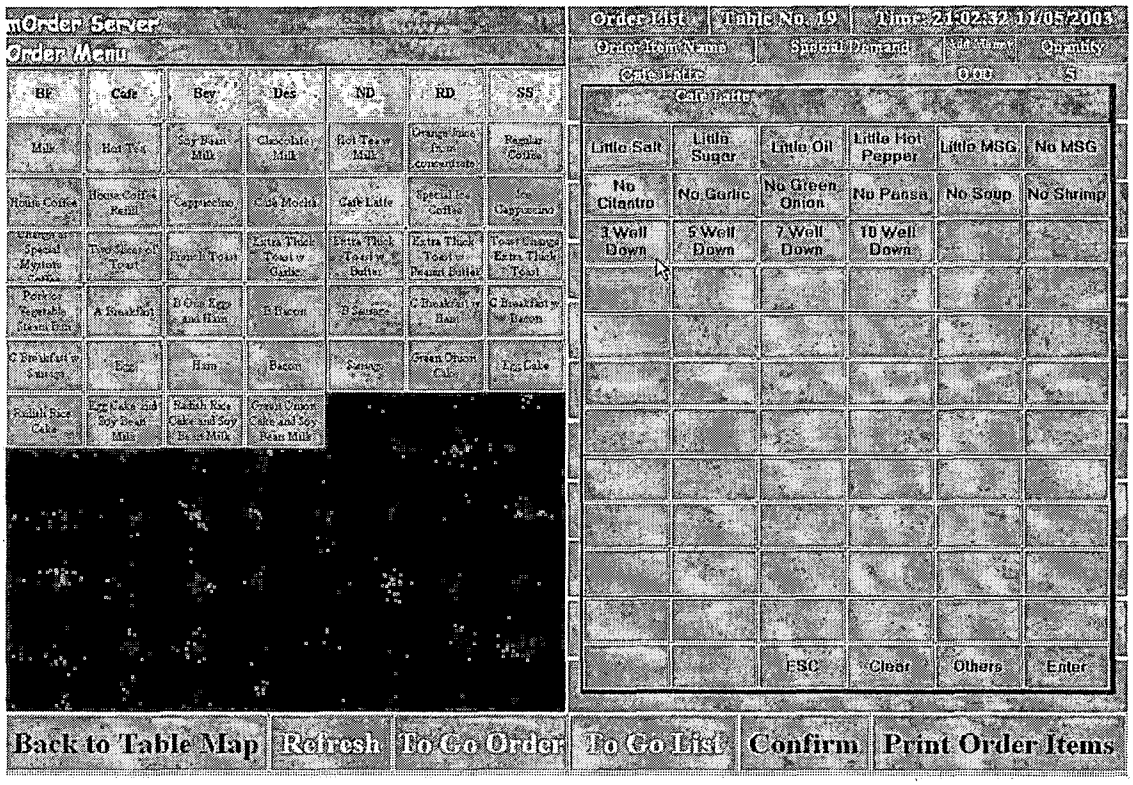


Figure 28. Special Demand_1

Click special demand options and the click "Enter".

Click "Esc" to exit special demand screen.

Click "Clear" to clear all selected special demand options.

Click "Others" to manually input special demand options and then click "Enter".

4.2.7 Add Money

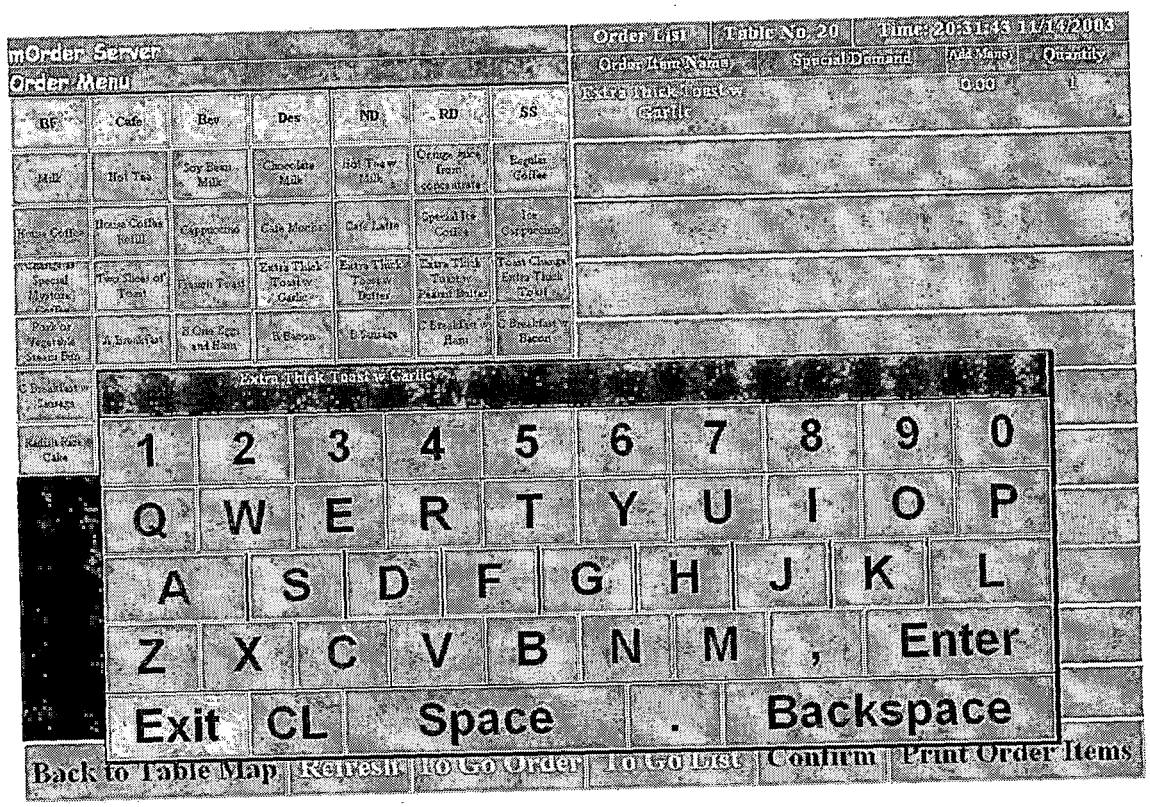


Figure 29. Add Money

Click "Add Money" option.

Input the amount money and click "Enter".

4.2.8 Cancel Hold

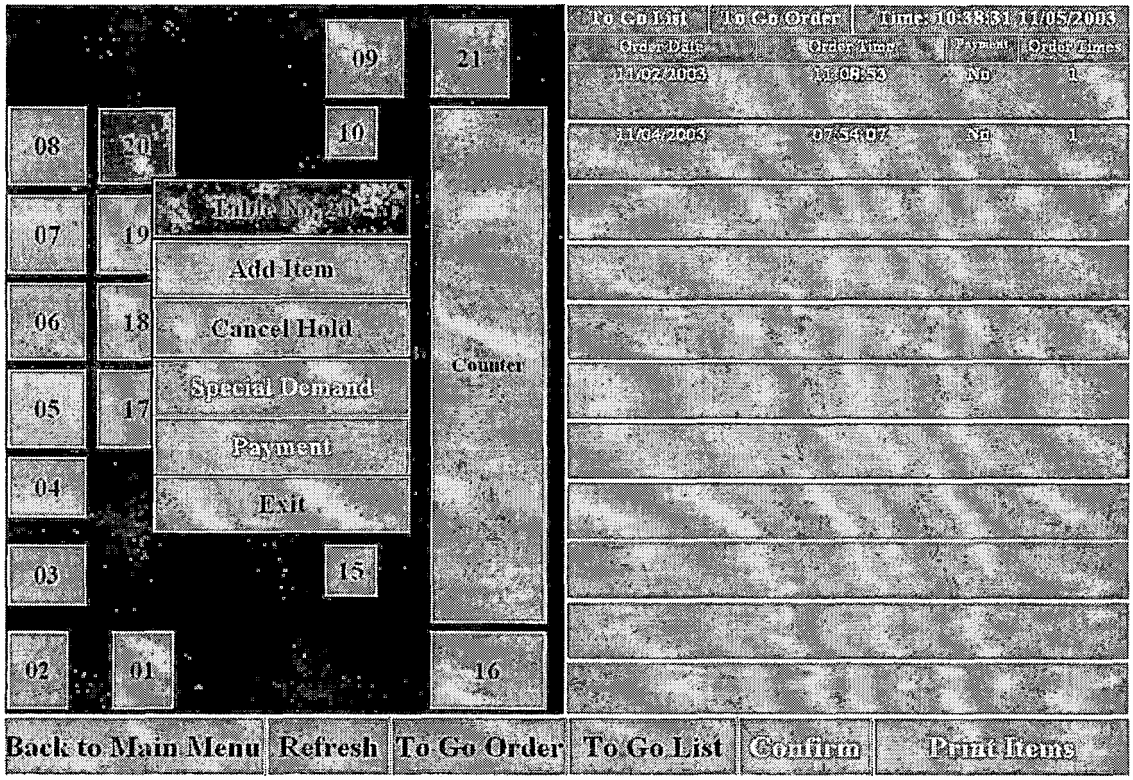


Figure 30. Cancel Hold

Click "Cancel Hold" button and the status of the table will become available.

4.2.9 Print Bill

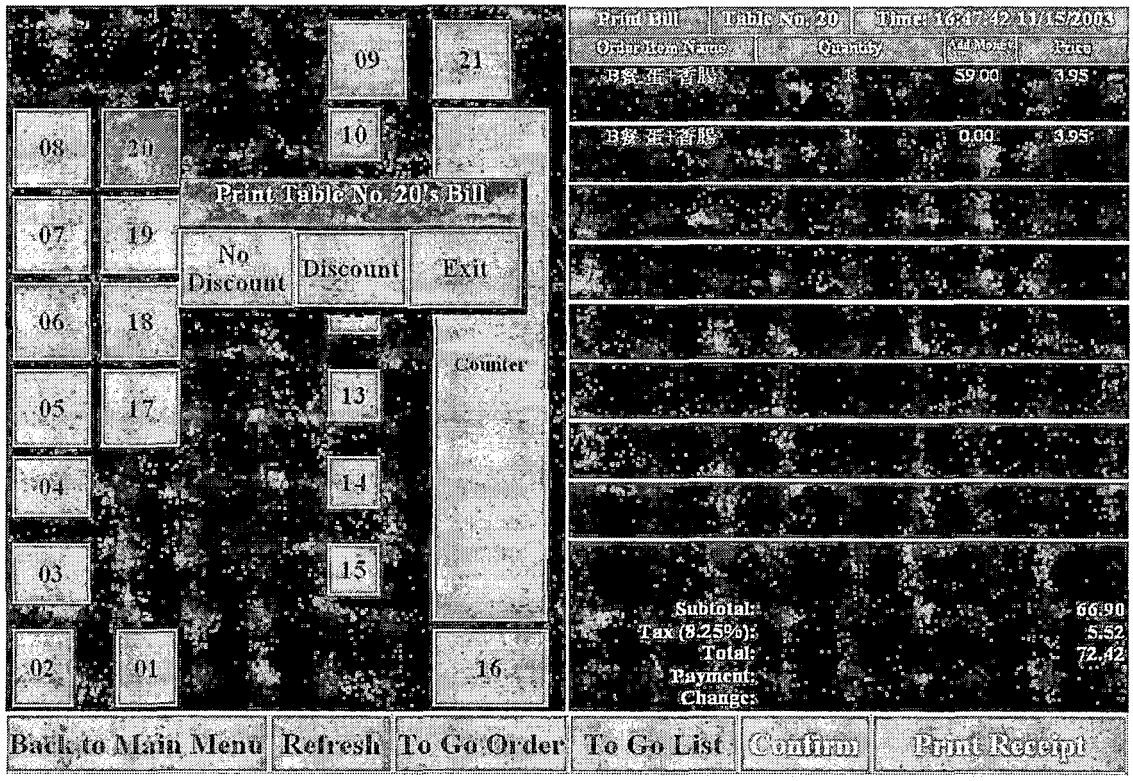


Figure 31. Print Bill

No Discount: Click "Print Bill".

4.2.10 Discount

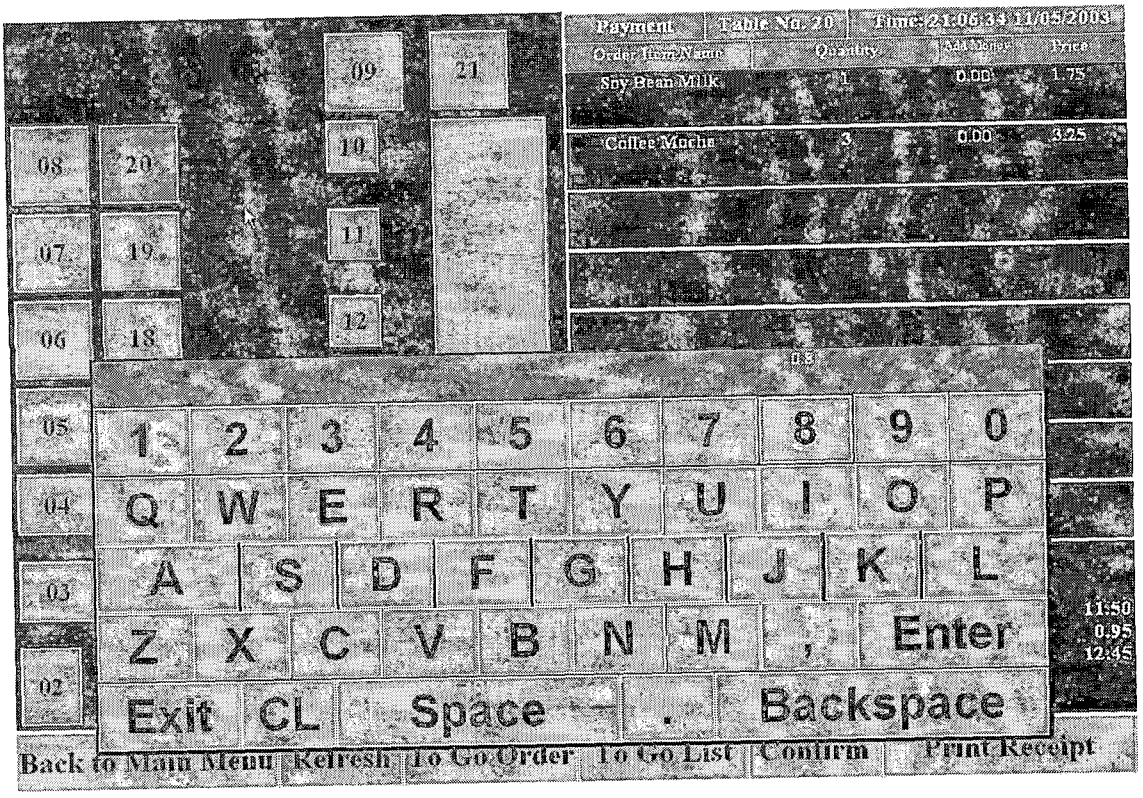


Figure 32. Discount

Discount: Click "Discount" and it will pop up a keyword. For example, if the discount is 20% off, input 0.8 and then click "Enter".

4.2.11 Order Item Status

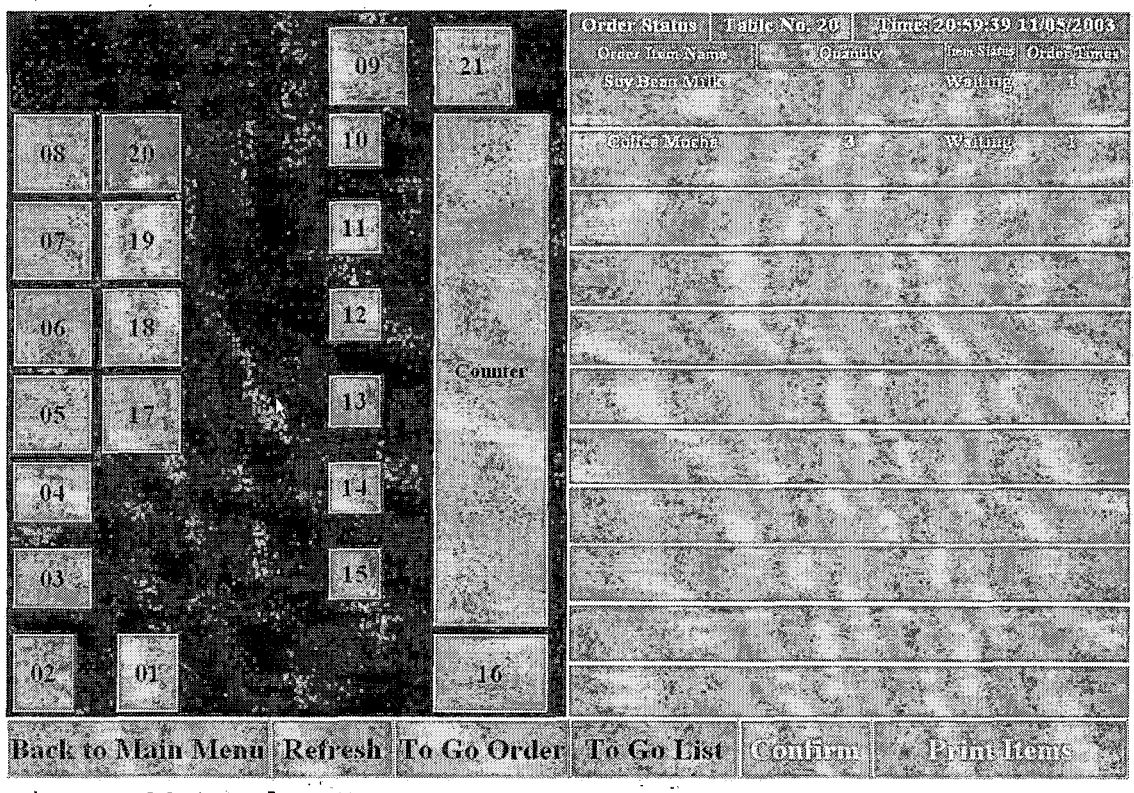


Figure 33. Order Item Status

Click "Order Item Status", and the right side will show the ordered item status.

4.2.12 Payment

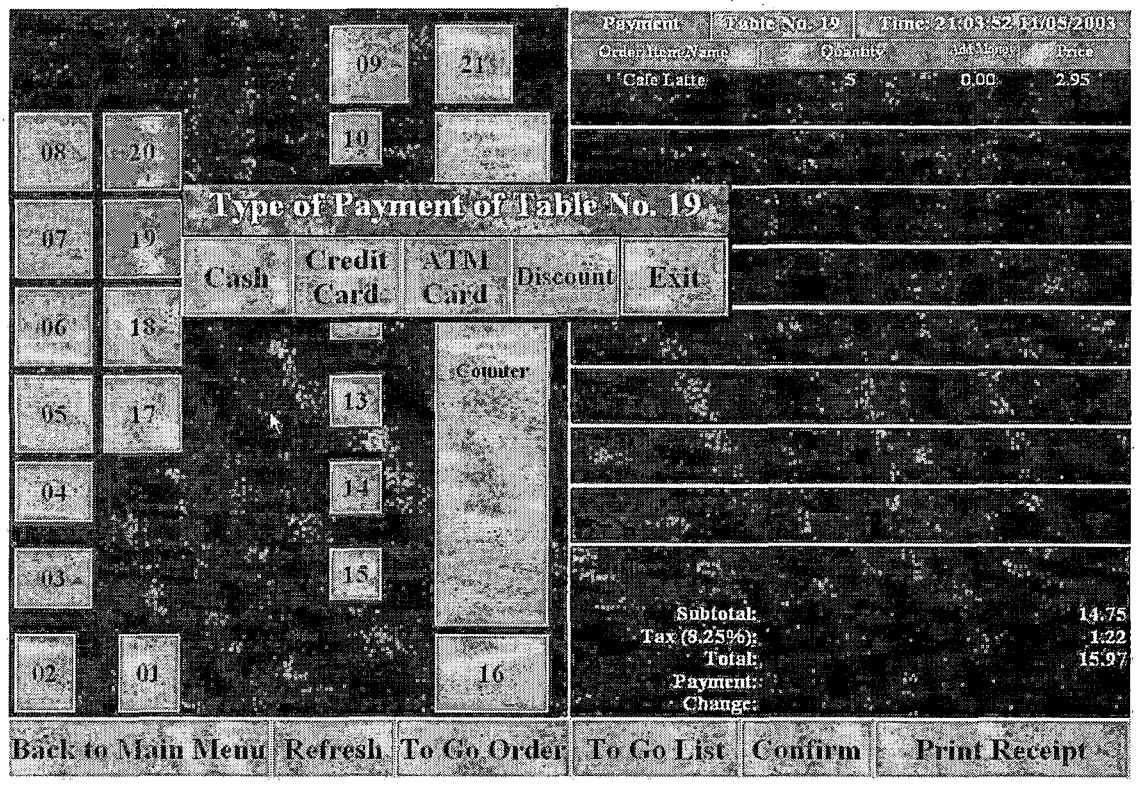


Figure 34. Payment

There are four options in the payment function.

Click "Cash".

4.2.12.1 Payment-Cash.

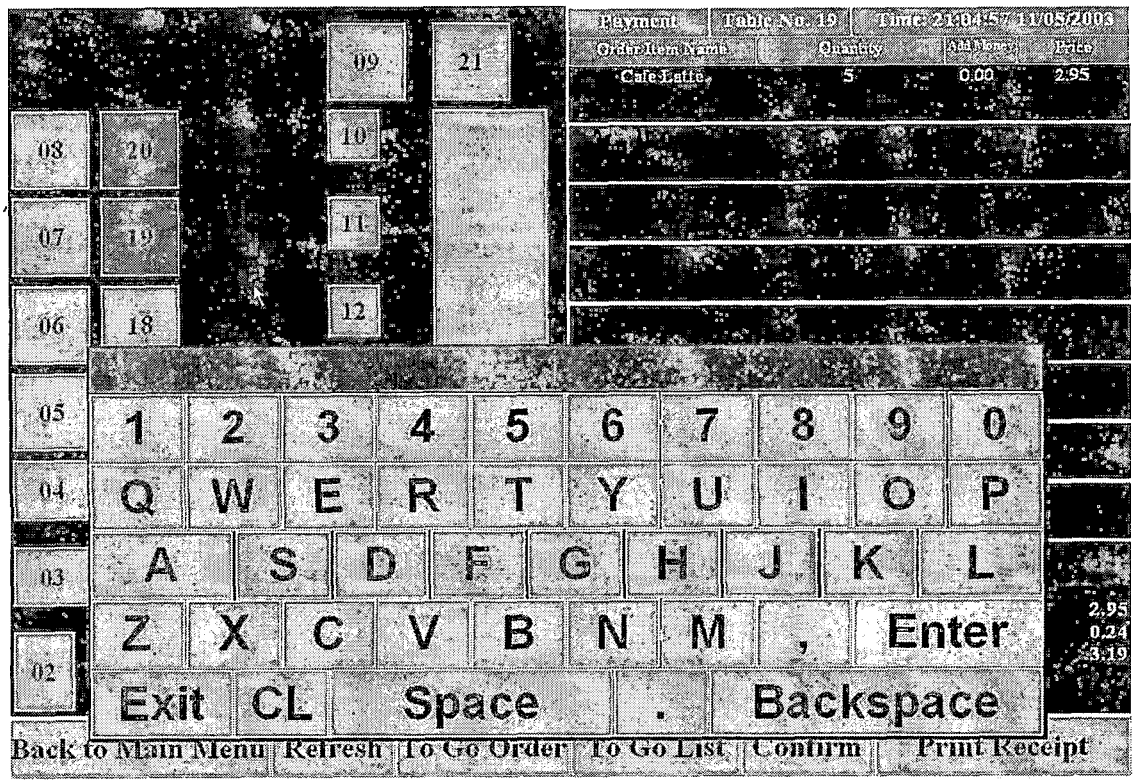


Figure 35. Payment-Cash

Input the amount that is greater or equal to the total.

Click "Enter".

Click "Confirm".

4.2.12.2 Payment-Credit or ATM.

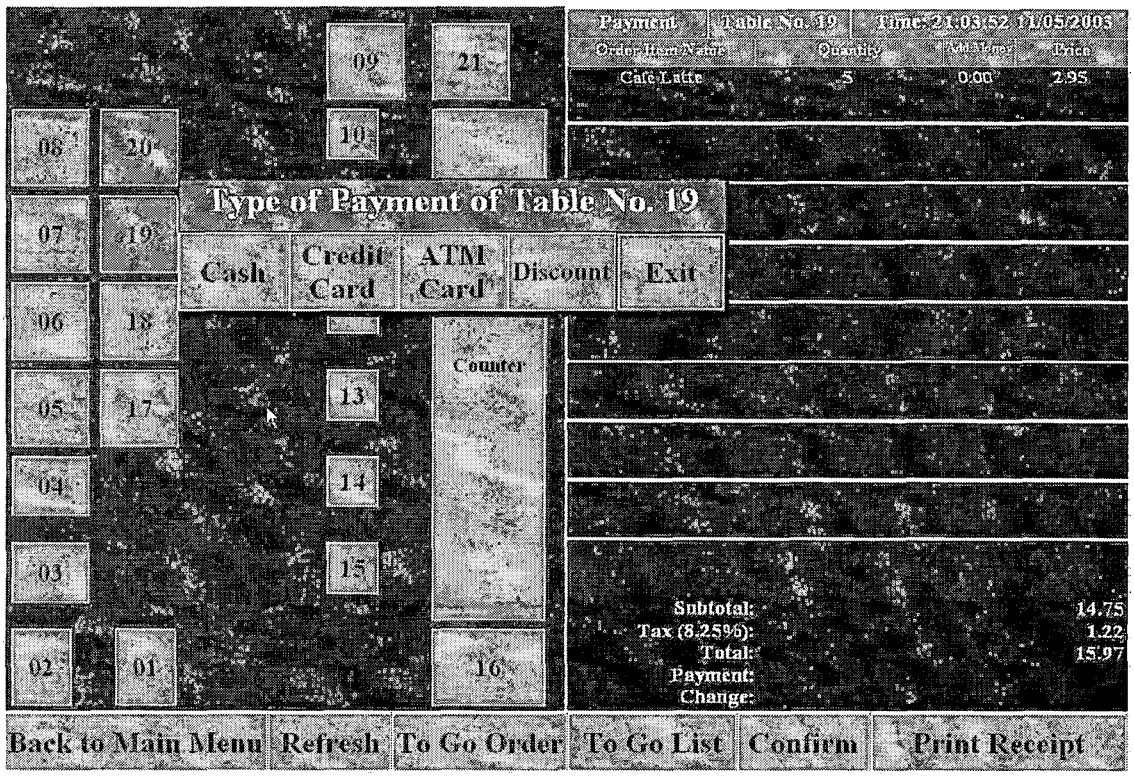


Figure 36. Payment-Credit or ATM

Click "Credit Card" or "ATM".

Click "Confirm".

4.2.12.3 Discount. Please refer to 4.2.10 Print Bill-Discount.

4.2.13 To Go Order

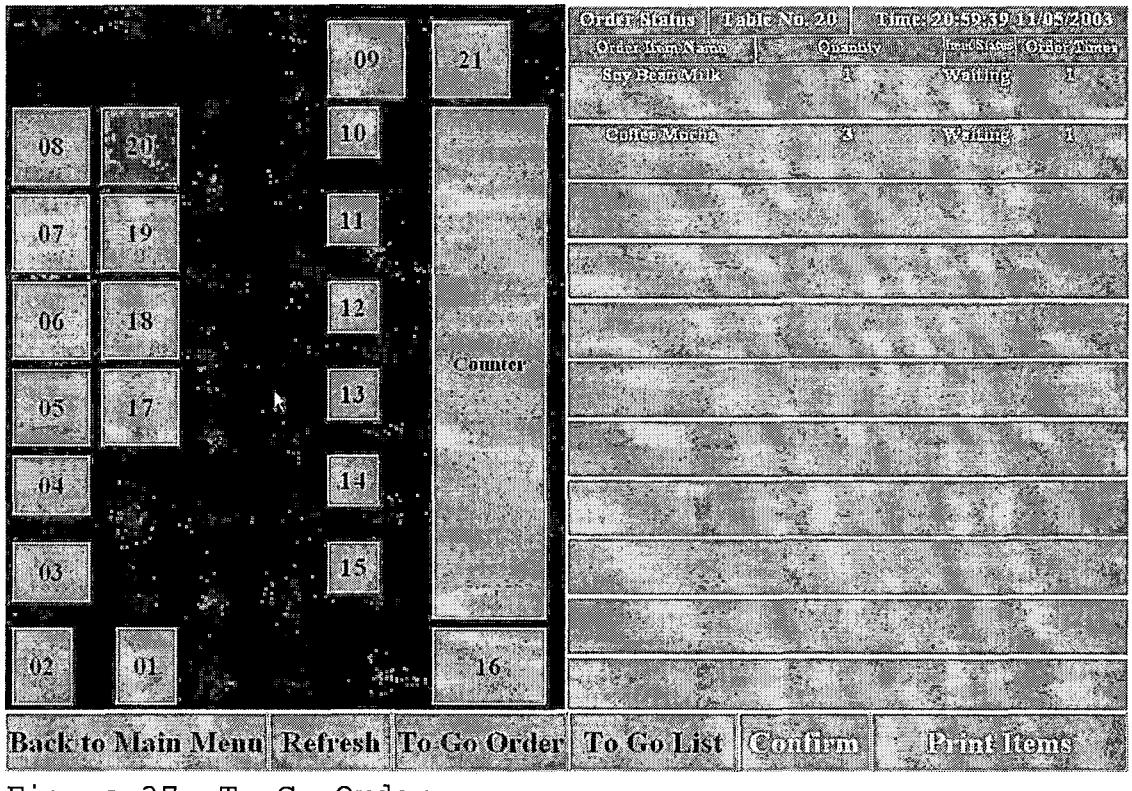


Figure 37. To Go Order

Click "To Go Order".

Please refer to "Menu", page 34.

4.2.14 To Go List

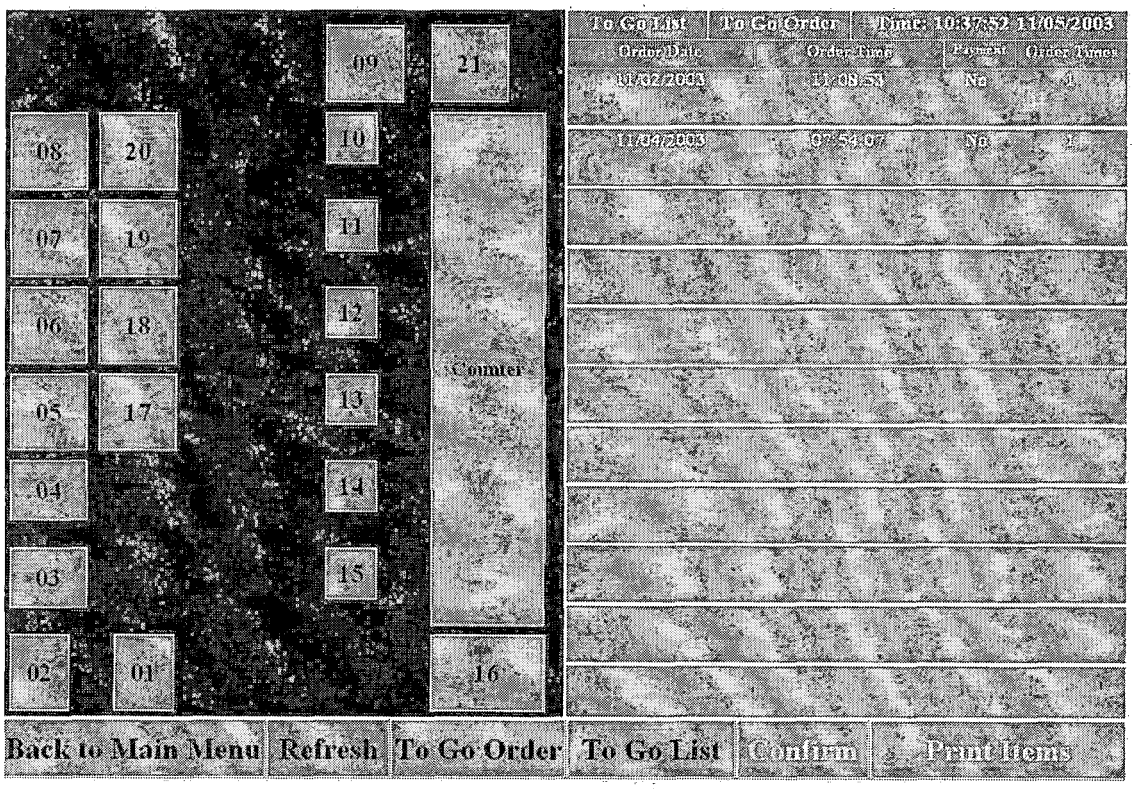


Figure 38. To Go List

Click "To Go List", and the right side will show all "To Go Orders".

4.2.15 Waitress Sales Report

Figure 39. Waitress Control Panel

In the waitress control panel, click "Sales Report".

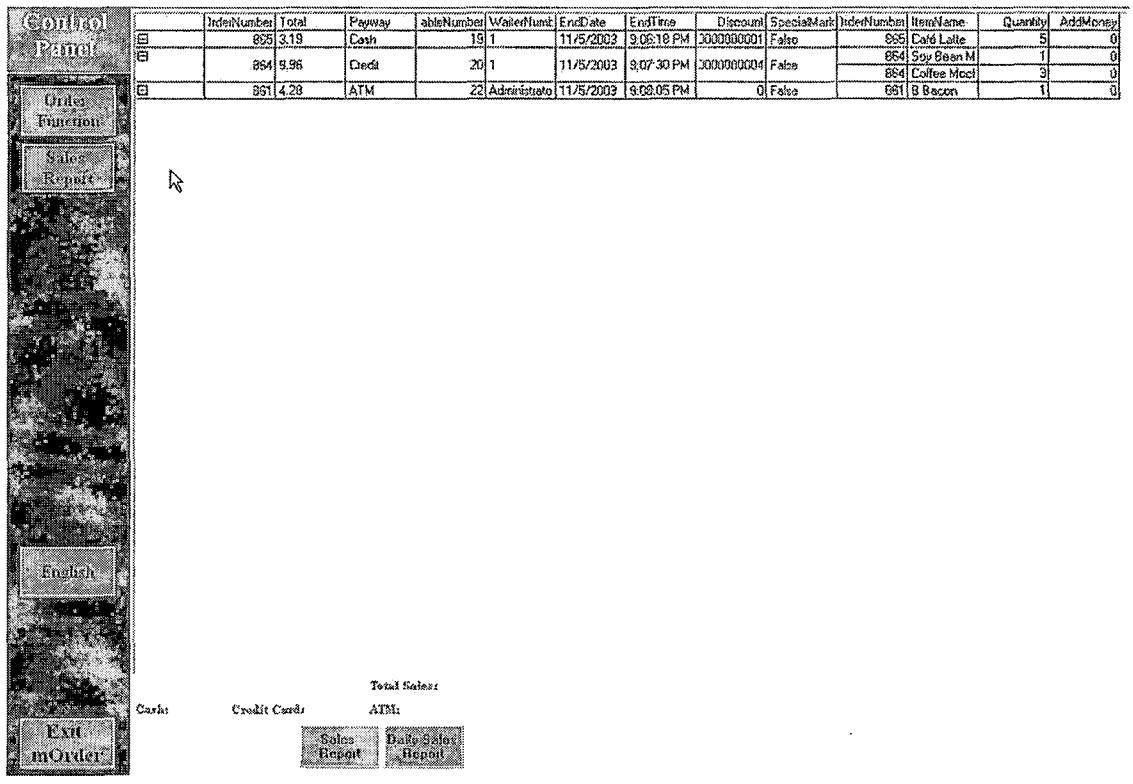


Figure 40. Waitress Sales Report

Click "Sales Report", it will show total sales during the period from the last closed time till now.

Click "Daily Sales Report", it will show total sales from the first order till now today.

4.3 Manager Function

4.3.1 Login

Figure 41. Manager Login

Click Login label and input "ADMINISTRATOR"

Click password label, input password and click "Enter".

4.3.2 Manager Control Panel

Figure 42. Manager Control Panel

There are 7 functions in the manager control panel. The order function is the same as waitress order function.

4.3.3 Manager Sales Report

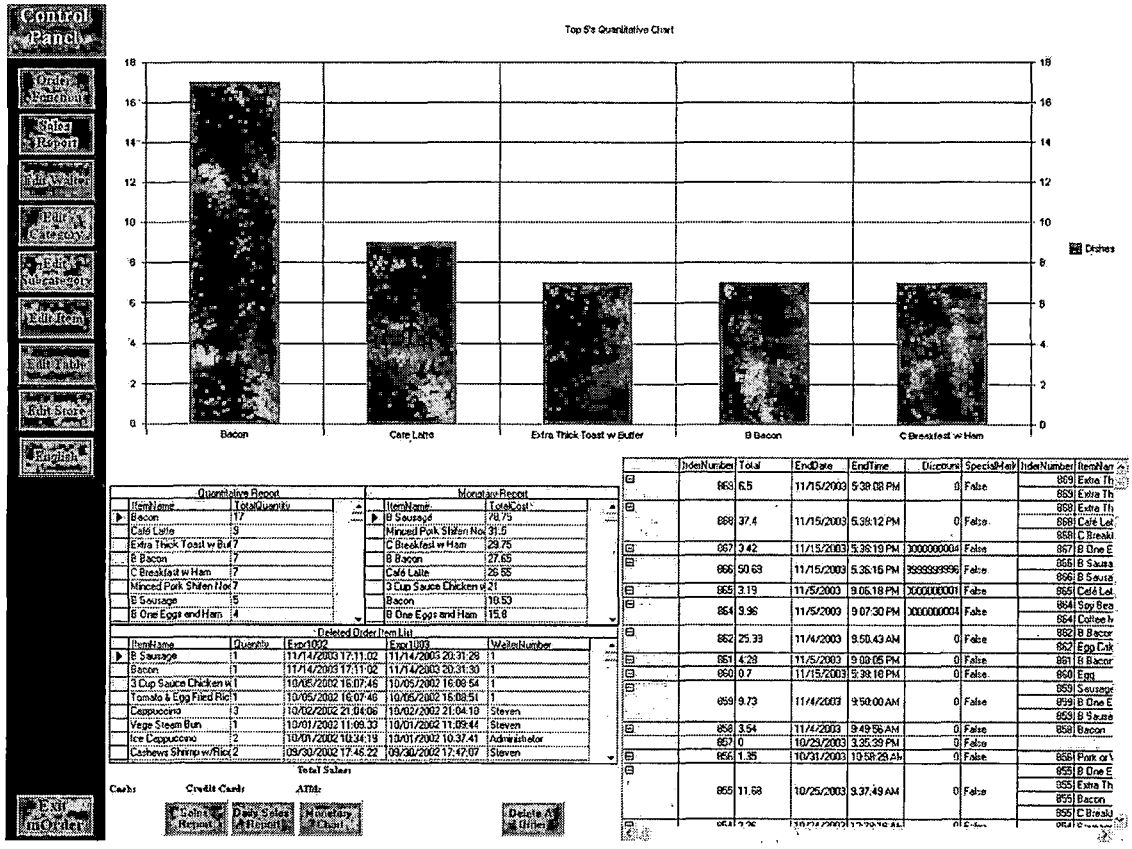


Figure 43. Manager Sales Report

Click "Sales Report".

"Sales Report" and "Daily Sales Report" are the same as waitress sales report.

Click "Monetary Chart" or "Quantitative Chart", the chart will show top 5 sales.

4.3.3.1 Delete Order History.

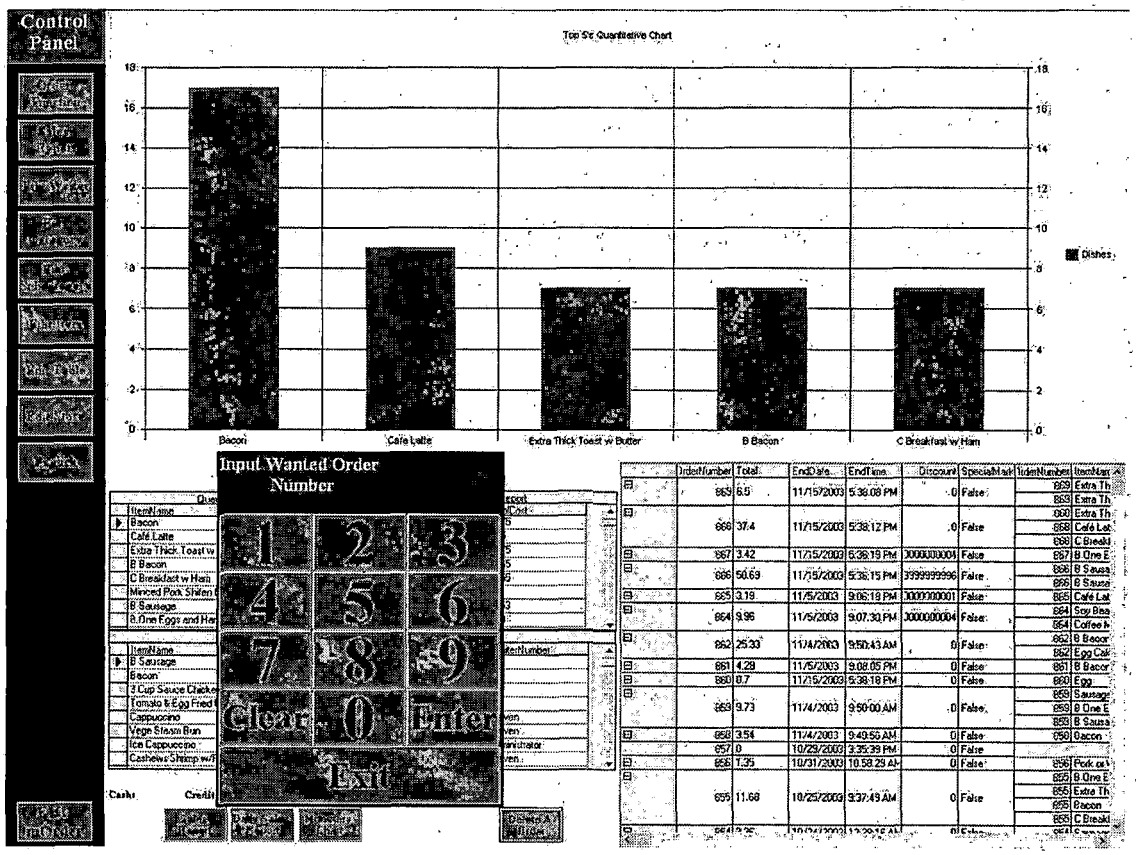


Figure 44. Delete Order History

Click "Delete a order".

Input the order number and click "Enter".

4.3.4 Edit Waiter

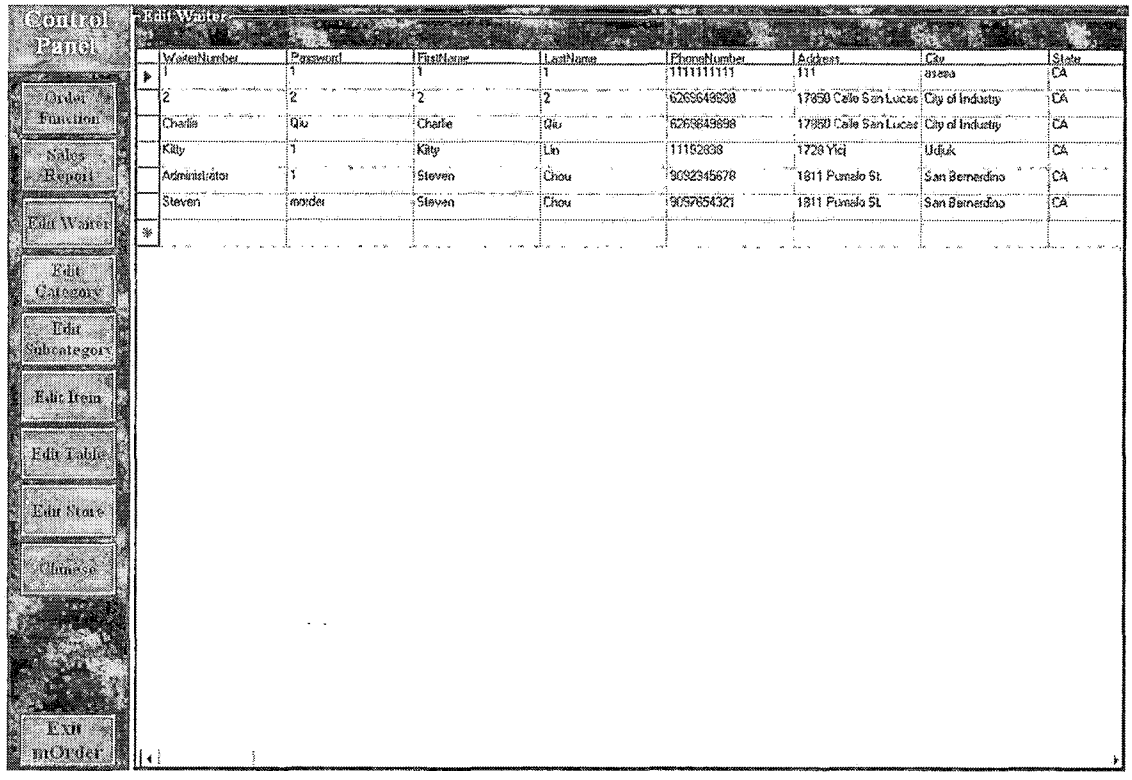


Figure 45. Edit Waiter

Click "Edit Waiter".

Enter the employee information in the last row. In the WaiterNumber column, the data must be unique and it can include number or text.

4.3.5 Edit Category

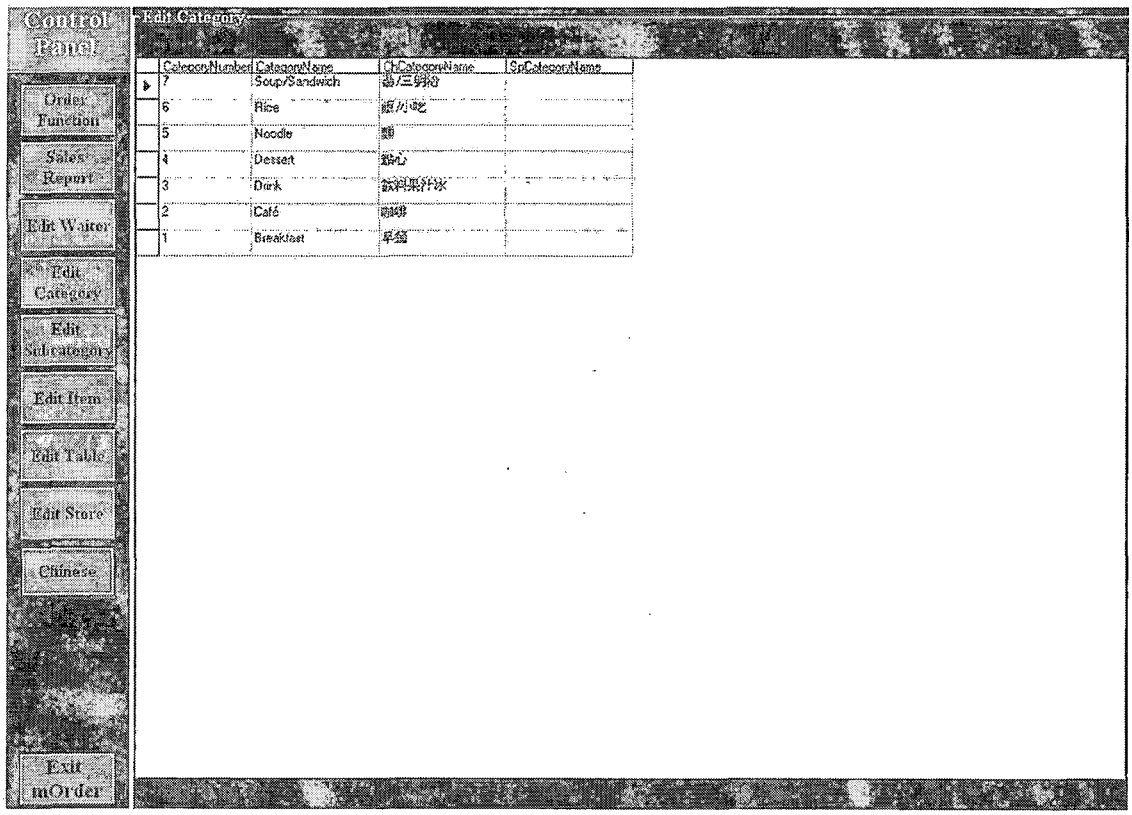


Figure 46. Edit Category

Click "Edit Category".

Edit "Category Name", "ChCategoryName", or "SpCategoryName".

4.3.6 Edit Sub Category-Add, Edit, Delete

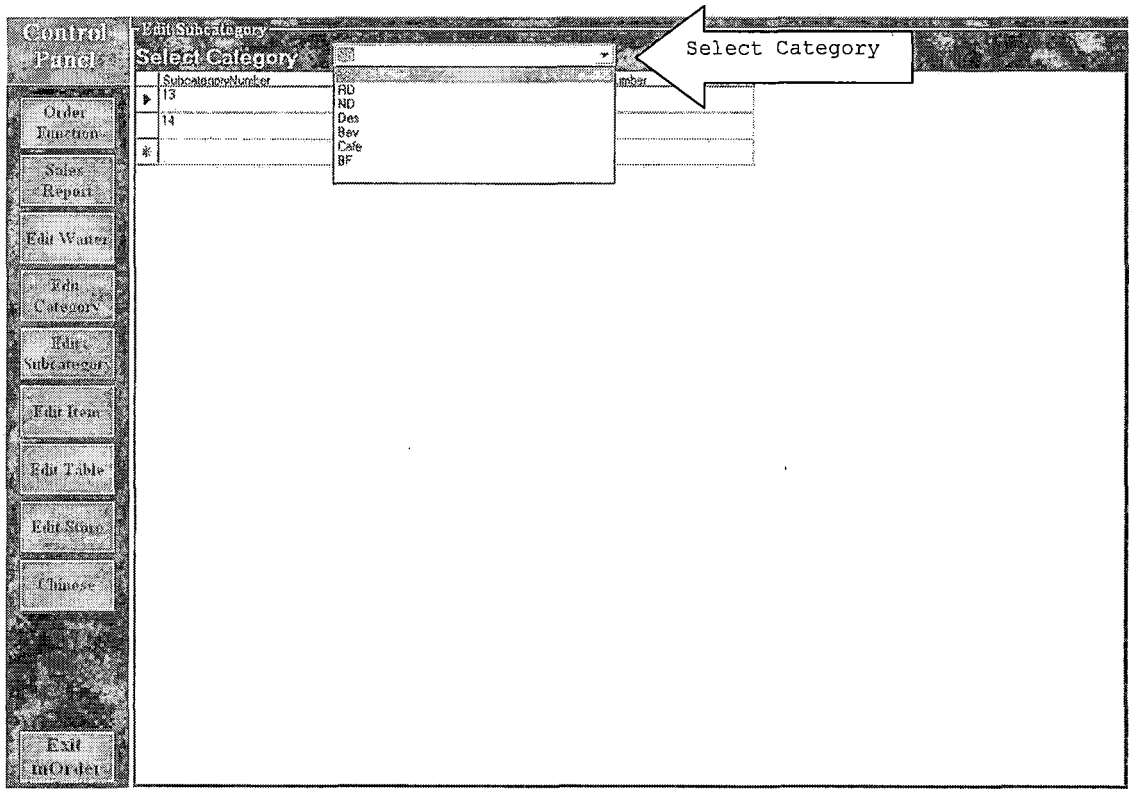


Figure 47. Edit Sub Category-Add, Edit, Delete

Add:

Select Category.

Enter "SubCategory" name and "Category" number. (Do not input any data in the SubCategoryNumber).

Edit:

Click the data and modify.

Delete

Select the row that want to delete.

Press "Delete" on the keyboard.

4.3.7 Edit Item-Add, Edit, Delete

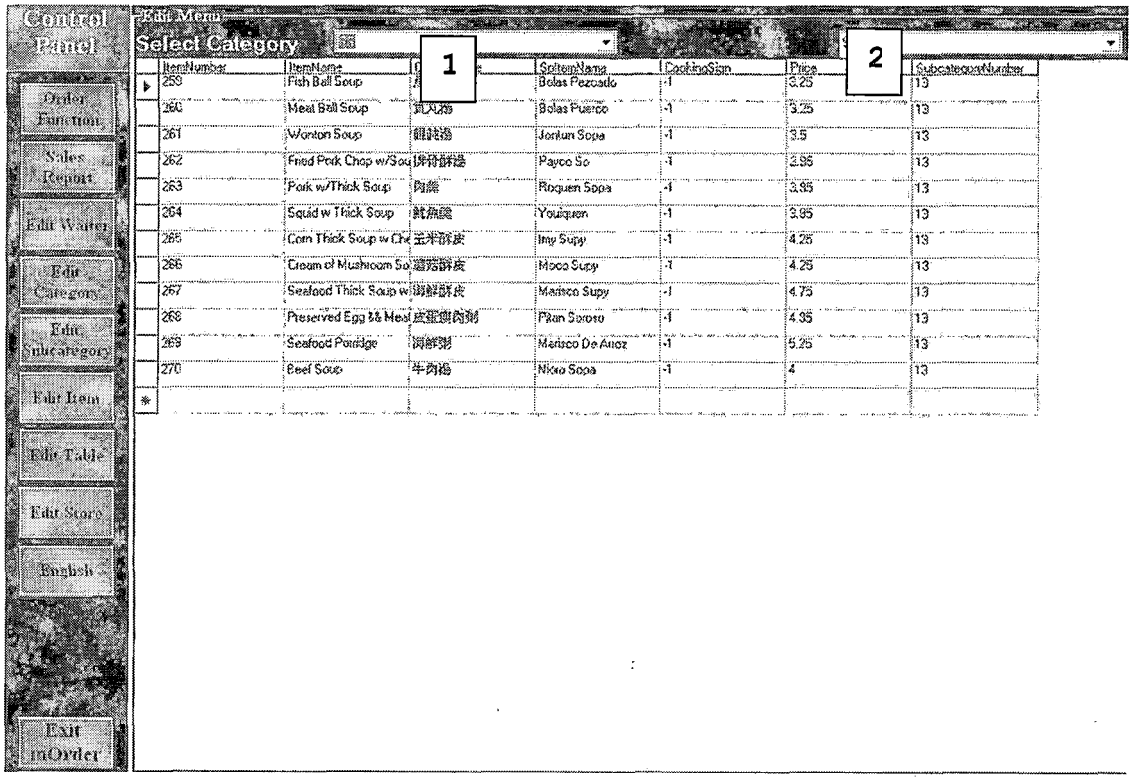


Figure 48. Edit Item-Add, Edit, Delete

Add:

Select Category.

Select Sub Category.

Input data expect "ItemNumber" in the last row.

Edit:

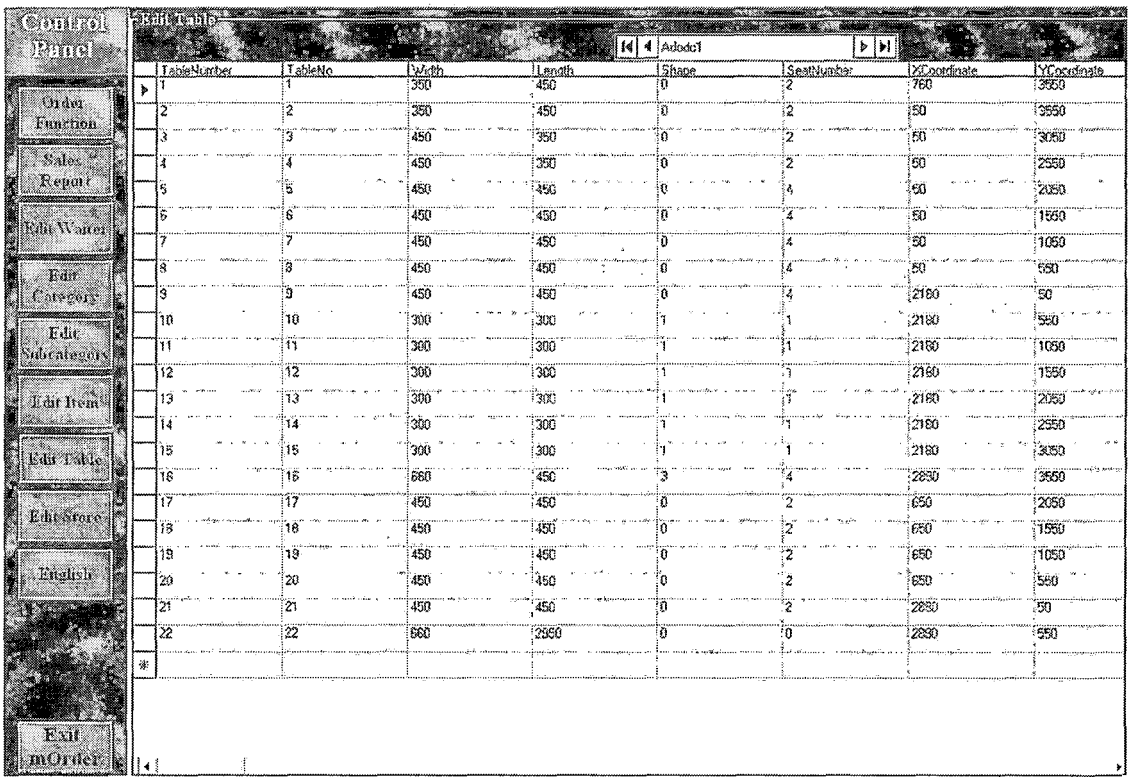
Click the data and modify.

Delete

Select the row that want to delete.

Press "Delete" on the keyboard.

4.3.8 Edit Table-Add, Edit, Delete



TableNumber	TableNo	Width	Length	Shape	SeatNumber	XCoordinate	YCoordinate
1	1	350	450	0	2	760	3550
2	2	350	450	0	2	50	3550
3	3	450	350	0	2	50	3050
4	4	450	350	0	2	50	2550
5	5	450	450	0	4	50	2050
6	6	450	450	0	4	50	1550
7	7	450	450	0	4	50	1050
8	8	450	450	0	4	50	550
9	9	450	450	0	4	2180	50
10	10	300	300	1	1	2180	550
11	11	300	300	1	1	2180	1050
12	12	300	300	1	1	2180	1550
13	13	300	300	1	1	2180	2050
14	14	300	300	1	1	2180	2550
15	15	300	300	1	1	2180	3050
16	16	660	450	3	4	2850	3550
17	17	450	450	0	2	650	2050
18	18	450	450	0	2	650	1550
19	19	450	450	0	2	650	1050
20	20	450	450	0	2	650	550
21	21	450	450	0	2	2850	50
22	22	660	2950	0	0	2850	550

Figure 49. Edit Table-Add, Edit, Delete

Add:

Input data in the last row.

Edit:

Click the data and modify.

Delete

Select the row that want to delete.

Press "Delete" on the keyboard.

4.3.9 Edit Store

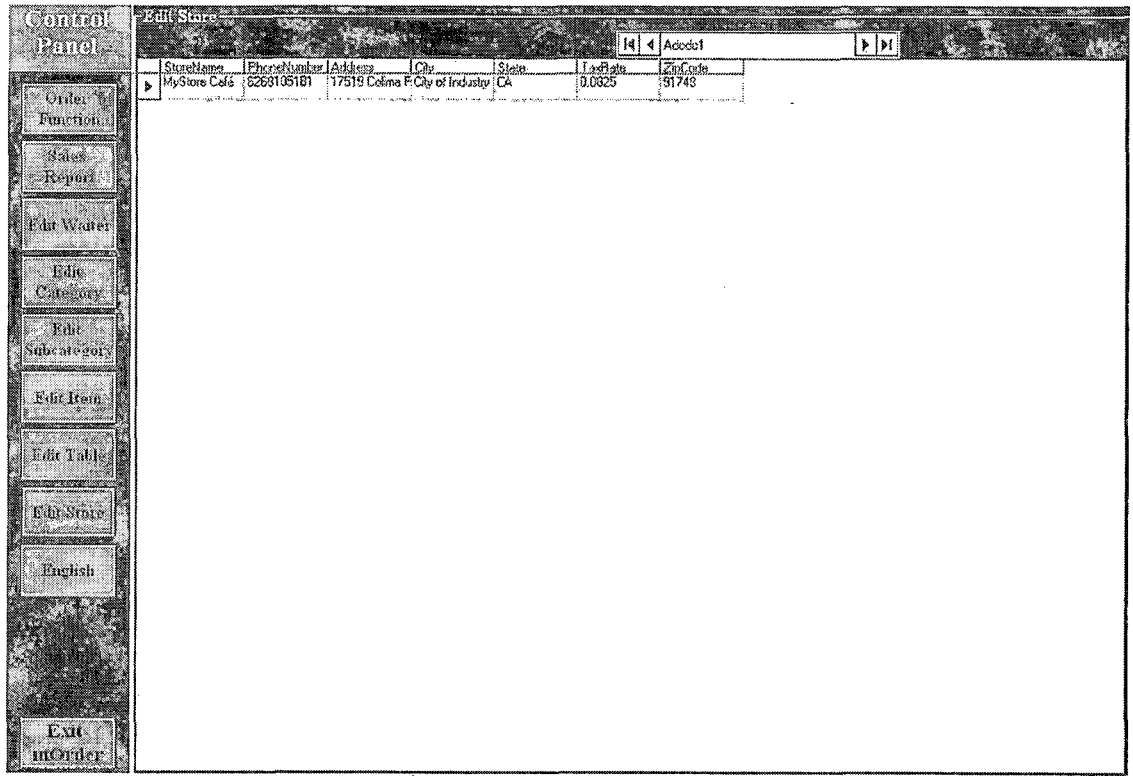


Figure 50. Edit Store

Click the data and modify.

CHAPTER FIVE

CONCLUSION

5.1 Summary

Because computers are becoming cheaper and cheaper, Point Of Sale systems are more and more popular among restaurants. There are many benefits to use POS instead of handwriting with cash register. For managers of restaurants, mOrder-Serevr helps them easily and quickly see sales statistics. It prevents stealing by employees and improves service for customers. For waitresses, mOrder-Server saves training time. Waitresses do not have to memorize all item numbers and prices. It saves time to serve customers. mOrder-Server with mOrder-Kitchen and Pocket PC, waitresses do not have to go back and forth between the counter and kitchen. It saves time to send ordered information through network. For cooks, mOrder-Server with mOrder-Kitchen makes cooks clearly know what meals they have not yet made. It prevents cooks from making wrong meals.

5.2 Future Developments

mOrder-Server is only for one restaurant environment. Now, a restaurant has more and more branches. In the future, mOrder-Server can integrate with Internet.

In a headquarter, managers can monitor all branches and edit item information. mOrder-Server will make it easy to manage restaurants.

APPENDIX

MORDER-SERVER SOURCE CODE

```
Const maxPPCNumber As Integer = 1
Const margin As Integer = 50
Const MaxTableNumber As Integer = 25
Const TableAvailableColor As Long = &H808000 '&HFFFF00  '&HC0C000
Const TableUnavailableColor As Long = &H404040
Const MaxFoodCategoryNumber As Integer = 10
Const MaxItemNumber As Integer = 100
Const ItemButtonHeight = 750 '13rows'812 ' 12 lines '950 '10 lines
Const MaxOrderItemNumber As Integer = 50
Const LabelHeight As Integer = 385
Const MaxSpecialDemandNumber As Integer = 68
Const MaxKeyNumber As Integer = 50
Const BeginningTime As String = "9:00:00"
Const MiddleTime As String = "17:00:00"
Const StopTime As String = "23:59:59"
Const WaitingCheckoutColor As Long = &H8080&
Const ButtonNonpressColor As Long = &H808000
Const BackgroundColor As Long = &HC00000
Const ButtonPressColor As Long = &H808080
Const CategoryButtonColor As Long = &H808080
Const CategoryButtonPressColor As Long = &H8080&  '&HC0C0&  '&HFFFF&
Const PrLanguageSwitch As String = "Spanish"
Const ReceiptLanguageSwitch As String = "English"
Const StartLanguageSwitch As String = "Chinese"
Const Administrator As String = "Administrator"

Dim editActionString As String
Dim cellNumber As Integer
Dim selectedItemString As String

Dim ItemButtonWidth As Integer
Dim orderItemButtonColor

Dim rPPCMessage() As String
Dim rkMessage As String

Dim Peedy As IAgentCtlCharacterEx
'---------------------- Printer ----------------------------------------
Dim sPrinter As New StarPrinter
Dim sSQLQuery, oConn, rs
Dim dbEngine As String
Dim stringLength As Integer
Dim i, j, k, l As Integer

Dim itemName As String
Dim itemQuantity As String
Dim itemProduct As Double
Dim addMoney As Double
Dim subtotal As Double
Dim tax As Double
Dim total As Double
Dim tNumber As String
Dim WaiterNumber As String
Dim Password As String
Dim wNumber As String

Dim storeName As String
Dim phoneNumber As String
Dim Address As String
Dim City As String
Dim State As String
Dim zipCode As String
Dim taxRate As Double

Dim totalTableNumber As Integer
Dim tableNumber() As Integer
Dim tableNo() As Integer
```

```
Dim tableAvailableStatus() As Boolean
Dim tableSeatNumber() As Integer
Dim tableXCoordinate() As Long
Dim tableYCoordinate() As Long
Dim tableWidth() As Long
Dim tableLength() As Long
Dim tableShape() As Integer
Dim xRate, yRate As Double
Dim tableNumberString As String
Dim str0, str1, str2, str3, str4, str5, str6 As String
Dim orderTableNumber As Integer

'Dim itemAddTimes As String

Dim CategoryName() As String
Dim totalCategoryNumber As Integer
Dim totalItemNumbers() As Integer
Dim itemNumbers() As String
Dim itemNames() As String
Dim itemPrices() As Double

Dim categoryIndex, itemIndex As Integer
Dim orderItemIndex As Integer

Dim orderNumber As String
Dim OrderItemNumber() As String
Dim orderItemAddMoney() As Double
Dim orderItemPromotion() As Double
Dim orderItemName() As String
Dim orderItemQuantity() As Integer
Dim orderItemSpecialDemand() As String
Dim orderSpecialDemand As String
Dim alreadyOrderItemNumber As Integer
Dim totalOrderItemNumber As Integer
Dim choiceItemName As String
Dim choiceSeriesNumber As String
Dim addTimes As Integer

Dim KeyboardNumber(10, 4) As String
Dim length As Integer
Dim actionSign As String

Dim startTime As String
Dim endTime As String
Dim timeString As String

Dim readOrderItemNumber As Integer
Dim maxHeight As Long

Dim totalCost As String

Dim totalSales As Double
Dim top5SalesName(5) As String
Dim top5SalesIncome(5) As Double

Dim discount As Double

Dim counterNumber As Integer
Dim counterTableNumber As Integer

Dim languageSwitch As String
Dim visibleStatus As Boolean

Dim delOrderNumber As String
Dim delItemNumber As String
Dim delQuantity As String
Dim delAddMoney As String
```

```
Dim delAddDate As String
Dim delAddTime As String
Dim delTotalCost As String

Dim prSpecialDemand(100) As String
Dim prOrderItemSpecialDemand() As String
Dim prSPFrameInputLabel As String

Dim Top20SalesArray()
Const Top20Sales As Integer = 5

Dim specialMark As Boolean
Dim bigSales As String
Dim smallSales As String

Dim itemLanguage As String
```

```
'StarPrinter-1

Public Sub OpenPrinter()
  Open "lpt1" For Output As #1
End Sub

Public Sub Center()
  Print #1, Chr$(&H1B); Chr$(&H1D); Chr$(&H61); Chr$(&H31);
End Sub

Public Sub LeftJustification()
  Print #1, Chr$(&H1B); Chr$(&H1D); Chr$(&H61); Chr$(&H30);
End Sub

Public Sub FontSize(s As Integer)
  Print #1, Chr$(&H1B); Chr$(&H57); Chr(s); Chr$(&H1B); Chr$(&H68); Chr$(s);
End Sub

Public Sub ClosePrinter()
  Print #1, Chr$(&H1B); Chr$(&H64); Chr$(&H32)
  Close #1
End Sub

Private Sub AddButton_Click()
  If AddButton.Caption = "Add Item" Then
    totalOrderItemNumber = 0
    If orderNumber = "" Then
      alreadyOrderItemNumber = 0
    Else
      ReadAlreadyOrderItemNumber
    End If
    ShowTableMapFrame False
    ShowMenuFrame True
    ClearOrderItemLabels
    ShowOrderItemFrame True
  ElseIf AddButton.Caption = "Modify Quantity" Then
    ShowNumericalFrame True
  End If
  ShowStatusFrame False
End Sub
```

```
Private Sub AddMoneyButton_Click()
  If AddMoneyButton.Caption = "Add Money" Then
    NumericalFrameTitle.Caption = OrderItemNameLabel(orderItemIndex)
    actionSign = AddMoneyButton.Caption
    ShowKeyboardFrame True
  ElseIf AddMoneyButton.Caption = "Order Item Status" Then
  ElseIf AddMoneyButton.Caption = "Payment" Then
    ShowPaymentFrame True
    mOrderForm.Caption = AddMoneyButton.Caption
    If Mid(StatusTitleButton.Caption, 1, 5) <> "Table" Then
      TableNoLabel.Caption = tableNumberString
    End If
    ClearOrderItemLabels
    ReadOrderItem2Checkout
    ShowOrderItemFrame True
  End If
  ShowStatusFrame False
End Sub

Private Sub AfternoonButton_Click()
  If Not NumericalFrame.Visible Then
    totalSales = 0
    startTime = "12:00:00"
    endTime = "16:59:59"
    ReportLabel.Caption = "Afternoon Sales Report"
    ReadAllOrderItems
  End If
End Sub

Private Sub CancelButton_Click()
  If CancelButton.Caption = "Cancel Hold" Then
    str2 = "True"
    str3 = tableNumber(orderTableNumber)
    str4 = ""
    UpdateTableavailableStatus
    ShowTable True
  ElseIf CancelButton.Caption = "Check Out" Then
    If mOrderForm.Caption = "Table Map" Then
      str2 = "True"
      str3 = tableNumber(orderTableNumber)
      str4 = ""
      UpdateTableavailableStatus
      ShowTable True
      Checkout
    Else
      'mOrderForm.Caption = "To Go List"
      tableNumberString = "To Go Order"
      Checkout
      ClearOrderItemLabels
      ReadToGoOrderInfo
      ShowOrderItemFrame True
    End If
  ElseIf CancelButton.Caption = "Cancel Item" Then
    OrderItemQuantityLabel(orderItemIndex).Caption = ""
    ShowOrderItems
    readOrderItemNumber = totalOrderItemNumber
    ShowOrderItemFrame True
  ElseIf CancelButton.Caption = "Cancel Ordered Item" Then
    DeleteOrderedItem
    mOrderForm.Caption = "Order Status"
    ClearOrderItemLabels
    ReadOrderItemStatus
    ShowOrderItemFrame True
    If delTotalCost <> "" Then
      ShowCheckoutFrame True
      ChangeLabel.Caption = delTotalCost
    End If
```

```
  ElseIf CancelButton.Caption = "Print Bill" Then
    mOrderForm.Caption = CancelButton.Caption
    ShowDiscountFrame True
    If Mid(StatusTitleButton.Caption, 1, 5) <> "Table" Then
      TableNoLabel.Caption = tableNumberString
    End If
    ClearOrderItemLabels
    ReadOrderItem2Checkout
    ShowOrderItemFrame True
  End If
  ShowStatusFrame False
End Sub

Private Sub CategoryButton_Click(Index As Integer)
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Then
  Else
    categoryIndex = Index
    For i = 1 To totalCategoryNumber
      CategoryButton(i).BackColor = CategoryButtonColor
    Next i
    CategoryButton(Index).BackColor = CategoryButtonPressColor
    ShowItem
  End If
End Sub

Private Sub Checkout()
  sSQLQuery = "UPDATE Orders SET EndDate='" + CStr(Date) + "', EndTime='" + CStr(Time) + _
        "' WHERE OrderNumber =" & orderNumber
  oConn.Open dbEngine
  oConn.Execute sSQLQuery
  oConn.Close
End Sub

Private Sub ClearOrderItemLabels()
    For i = 1 To MaxOrderItemNumber
      OrderItemNameLabel(i).Caption = ""
      OrderItemDemandLabel(i).Caption = ""
      PrOrderItemDemandLabel(i).Caption = ""
      OrderItemQuantityLabel(i).Caption = ""
      OrderItemNumberLabel(i).Caption = ""
      OrderItemAddMoneyLabel(i).Caption = ""
      OrderItemPromotionLabel(i).Caption = ""
    Next i
End Sub

Private Sub CategoryDataCombo_Change()
  sSQLQuery = "SELECT * FROM SubfoodCategory " + _
        "WHERE CategoryNumber IN (Select CategoryNumber FROM FoodCategory " + _
                "WHERE CategoryName='" + CategoryDataCombo.Text + "')"
  SubcategoryAdodc.RecordSource = sSQLQuery
  SubcategoryAdodc.Refresh
  If editActionString = "Edit Menu" Then
    sSQLQuery = "SELECT * FROM Item " + _
          "WHERE SubcategoryNumber IN (Select SubcategoryNumber FROM SubfoodCategory " + _
                "WHERE SubcategoryName='" + SubcategoryDataCombo.Text + "')"
    ItemAdodc.RecordSource = sSQLQuery
    ItemAdodc.Refresh
  End If
End Sub

Private Sub ConfirmButton_Click()
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Or SpecialMarkFrame.Visible Then
  Else
    If mOrderForm.Caption = "Payment" Then
      If PaymentLabel.Caption = "" Then
```

```
            ShowPaymentFrame True
          Else
            PaymentFunction
            'ShowSpecialMarkFrame True
          End If
        ElseIf mOrderForm.Caption = "Order Menu" Or mOrderForm.Caption = "Order List" Then
          If orderNumber = "" Then
            addTimes = 0
            GetOrderNumber
            InsertOrderItems2DB
          Else
            InsertOrderItems2DB
          End If
          'PrintOrderList orderNumber
          ShowMenuFrame False
          ShowTableMapFrame True
        End If
      End If
    End Sub

    Private Sub DaytimeButton_Click()
      If Not NumericalFrame.Visible Then
        totalSales = 0
        ReadAccountingInfo
        ReportLabel.Caption = "Sales Report " + "From " + startTime + " To " + endTime
        ReadAllOrderItems
        UpdateAccountingInfo
      End If
    End Sub

    Private Sub DeleteAOrder(wantedOrderNumber As String)
      sSQLQuery = "DELETE FROM Orders WHERE OrderNumber=" + wantedOrderNumber
      oConn.Open dbEngine
      oConn.Execute sSQLQuery
      oConn.Close
      OrderListAdodc.Refresh
      QuantitativeStatisticsAdodc.Refresh
      MonetaryStatisticsAdodc.Refresh
      DeletedOrderItemAdodc.Refresh
    End Sub

    Private Sub DeleteOrderedItem()
      sSQLQuery = "SELECT OrderNumber, ItemNumber, Quantity, AddMoney, AddDate, AddTime " + _
        "FROM OrderItem " + _
        "WHERE SeriesNumber=" + choiceSeriesNumber
      oConn.Open dbEngine
      rs.Open sSQLQuery, oConn
      delOrderNumber = rs("OrderNumber")
      delItemNumber = rs("ItemNumber")
      delQuantity = rs("Quantity")
      delAddMoney = rs("AddMoney")
      delAddDate = rs("AddDate")
      delAddTime = rs("AddTime")
      oConn.Close

      sSQLQuery = "DELETE FROM OrderItem WHERE SeriesNumber=" + choiceSeriesNumber
      oConn.Open dbEngine
      oConn.Execute sSQLQuery
      oConn.Close

      sSQLQuery = "INSERT INTO DeletedOrderItem(OrderNumber, ItemNumber, Quantity, " + _
                "AddMoney, AddDate, AddTime, DeletedDate, DeletedTime, WaiterNumber) " + _
            "VALUES (" + delOrderNumber + "," + delItemNumber + "," + delQuantity + "," + _
                delAddMoney + ",'" + delAddDate + "','" + delAddTime + "','" + _
                CStr(Date) + "','" + CStr(Time) + "','" + _
                WaiterNumber + "')"
      oConn.Open dbEngine
```

```
  oConn.Execute sSQLQuery
  oConn.Close

'--------------------- Already pay money -----------------------------------
On Error Resume Next
  totalCost = ""
  discount = 0#
  delTotalCost = ""
  sSQLQuery = "SELECT Total, Discount FROM Orders WHERE OrderNumber=" + delOrderNumber
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  totalCost = rs("Total")
  discount = rs("Discount")
  oConn.Close

  If totalCost <> "" Then
    sSQLQuery = "SELECT Price FROM Item WHERE ItemNumber=" + delItemNumber
    oConn.Open dbEngine
    rs.Open sSQLQuery, oConn
    delTotalCost = FormatNumber(delQuantity * (rs("Price") + delAddMoney) * (1 + taxRate), 2, vbTrue)
    totalCost = totalCost - delTotalCost
    oConn.Close

    sSQLQuery = "UPDATE Orders SET Total='" + FormatNumber(totalCost, , vbTrue, , vbFalse) + _
          "' WHERE OrderNumber =" & delOrderNumber
    oConn.Open dbEngine
    oConn.Execute sSQLQuery
    oConn.Close

    SubtotalLabel.Caption = ""
    TaxLabel.Caption = ""
    TotalLabel.Caption = ""
    PaymentLabel.Caption = ""
  End If
End Sub

Private Sub DeleteAOrderButton_Click()
  If Not NumericalFrame.Visible And WaiterNumber = Administrator Then
    mOrderForm.Caption = "Sales Report"
    ShowNumericalFrame True
  End If
End Sub

Private Sub DemandButton_Click()
  If DemandButton.Caption = "Order Item Status" Then
    mOrderForm.Caption = "Order Status"
    ClearOrderItemLabels
    ReadOrderItemStatus
    ShowOrderItemFrame True
  ElseIf DemandButton.Caption = "Special Demand" Then
    ShowSPFrame True
  End If
  ShowStatusFrame False
End Sub

Private Sub DiscountButton_Click()
  actionSign = "Discount Print"
  ShowKeyboardFrame True
  ShowDiscountFrame False
  mOrderForm.Caption = CancelButton.Caption
  If Mid(StatusTitleButton.Caption, 1, 5) <> "Table" Then
    TableNoLabel.Caption = tableNumberString
  End If
  ClearOrderItemLabels
  ReadOrderItem2Checkout
  ShowOrderItemFrame True
End Sub
```

```
Private Sub DiscountTitle_Click()
  'ShowDiscountFrame False
End Sub

Private Sub EditCategory_Click()
  If WaiterNumber = Administrator And Not NumericalFrame.Visible Then
    CategoryAdodc.Refresh
    editActionString = "Edit Category"
    ShowEditFrame True
    ShowOrderFrame False
    ShowSalesReportFrame False
  End If
End Sub

Private Sub EditMenuButton_Click()
  If WaiterNumber = Administrator And Not NumericalFrame.Visible Then
    editActionString = "Edit Menu"
    ShowEditFrame True
    ShowOrderFrame False
    ShowSalesReportFrame False
  End If
End Sub

Private Sub EditStoreButton_Click()
  If WaiterNumber = Administrator And Not NumericalFrame.Visible Then
    editActionString = "Edit Store"
    ShowEditFrame True
    ShowOrderFrame False
    ShowSalesReportFrame False
  End If
End Sub

Private Sub EditSubcategory_Click()
  If WaiterNumber = Administrator And Not NumericalFrame.Visible Then
    editActionString = "Edit Subcategory"
    ShowEditFrame True
    ShowOrderFrame False
    ShowSalesReportFrame False
  End If
End Sub

Private Sub EditTableButton_Click()
  If WaiterNumber = Administrator And Not NumericalFrame.Visible Then
    editActionString = "Edit Table"
    ShowEditFrame True
    ShowOrderFrame False
    ShowSalesReportFrame False
  End If
End Sub

Private Sub EditWaiterButton_Click()
  If WaiterNumber = Administrator And Not NumericalFrame.Visible Then
    editActionString = "Edit Waiter"
    ShowEditFrame True
    ShowOrderFrame False
    ShowSalesReportFrame False
  End If
End Sub

Private Sub ExitButton_Click()
  If Not NumericalFrame.Visible Then
    'ExitFrame.Visible = True
    'No.SetFocus
    'End
    ShowLoginFrame True
    ShowSalesReportFrame False
```

```
    ShowOrderFrame False
    ShowEditFrame False
  End If
End Sub

Private Sub ExitOrderButton_Click()
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Or SpecialMarkFrame.Visible Then
  Else
    If mOrderForm.Caption = "Order List" Or mOrderForm.Caption = "Order Menu" Then
      ShowTableMapFrame True
    ElseIf mOrderForm.Caption = "Table Map" Or mOrderForm.Caption = "Check Out" Or _
        mOrderForm.Caption = "Payment" Or mOrderForm.Caption = "Print Bill" Or _
        mOrderForm.Caption = "Order Status" Or mOrderForm.Caption = "To Go List" Then
      ShowOrderFrame False
      ShowTableMapFrame False
    End If
    ShowStatusFrame False
    ShowMenuFrame False
    ShowNumericalFrame False
    ShowSPFrame False
    ShowPaymentFrame False
    ShowKeyboardFrame False
  End If
End Sub

Public Function FixedLengthString(aString As String, anInteger As String)
  stringLength = Len(aString)
  If stringLength >= anInteger Then
    FixedLengthString = Mid(aString, 1, anInteger)
  Else
    FixedLengthString = aString
    stringLength = anInteger - stringLength
      For i = 1 To stringLength
        If anInteger > 8 Then
          FixedLengthString = FixedLengthString + " "
        Else
          FixedLengthString = " " + FixedLengthString
        End If
      Next i
  End If
End Function

Private Sub Form_Load()

  ReDim rPPCMessage(maxPPCNumber)

  ReDim tableNumber(MaxTableNumber)
  ReDim tableNo(MaxTableNumber)
  ReDim tableAvailableStatus(MaxTableNumber)
  ReDim tableSeatNumber(MaxTableNumber)
  ReDim tableXCoordinate(MaxTableNumber)
  ReDim tableYCoordinate(MaxTableNumber)
  ReDim tableWidth(MaxTableNumber)
  ReDim tableLength(MaxTableNumber)
  ReDim tableShape(MaxTableNumber)

  ReDim CategoryName(MaxFoodCategoryNumber)
  ReDim totalItemNumbers(MaxFoodCategoryNumber)
  ReDim itemNumbers(MaxItemNumber, MaxFoodCategoryNumber)
  ReDim itemNames(MaxItemNumber, MaxFoodCategoryNumber)
  ReDim itemPrices(MaxItemNumber, MaxFoodCategoryNumber)

  ReDim OrderItemNumber(MaxOrderItemNumber)
  ReDim OrderItemNumber(MaxOrderItemNumber)
  ReDim orderItemAddMoney(MaxOrderItemNumber)
  ReDim orderItemPromotion(MaxOrderItemNumber)
```

```
ReDim orderItemName(MaxOrderItemNumber)
ReDim orderItemQuantity(MaxOrderItemNumber)
ReDim orderItemSpecialDemand(MaxOrderItemNumber)
ReDim prOrderItemSpecialDemand(MaxOrderItemNumber)
ReDim listOrderItemNumber(MaxOrderItemNumber)
ReDim listOrderItemAddMoney(MaxOrderItemNumber)
ReDim listOrderItemPromotion(MaxOrderItemNumber)

ReDim specialDemandNumber(MaxSpecialDemandNumber)
ReDim specialDemandString(MaxSpecialDemandNumber)

ReDim Top20SalesArray(1 To Top20Sales, 1 To 2)

taxRate = 0.078
If StartLanguageSwitch = "Chinese" Then
  languageSwitch = "Chinese"
  LanguageButton.Caption = "Chinese"
  itemLanguage = "ChItemName"
Else
  languageSwitch = "English"
  LanguageButton.Caption = "English"
  itemLanguage = "ItemName"
End If

For i = 1 To MaxTableNumber
  Load TableButtons(i)
Next i

If mOrderForm.WindowState = Normal Then
  Move Screen.Width - Width, Screen.Height - Height
End If
Set oConn = CreateObject("ADODB.Connection")
Set rs = CreateObject("ADODB.Recordset")
dbEngine = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\morder\mOrderDatabase.mdb"

ESC = Chr$(27)
Cutter = Chr$(100)
lineFeed = Chr$(10)

Agent1.Characters.Load "Peedy", DATAPATH
Set Peedy = Agent1.Characters("Peedy")
Peedy.LanguageID = &H409


For i = 1 To maxPPCNumber
  Load PPCSocket(i)
  PPCSocket(i).LocalPort = ServerIPPort
  PPCSocket(i).Listen
Next i

KitchenSocket.LocalPort = kPCIPPort
KitchenSocket.Listen

ReadFoodCategory
ReadItemInfo

For i = 1 To totalCategoryNumber
  Load CategoryButton(i)
Next i
For i = 1 To MaxItemNumber
  Load ItemButton(i)
Next i

For i = 1 To MaxItemNumber
  Load OrderItemNameLabel(i)
  Load OrderItemDemandLabel(i)
  Load PrOrderItemDemandLabel(i)
```

```
    Load OrderItemQuantityLabel(i)
    Load OrderItemNumberLabel(i)
    Load OrderItemAddMoneyLabel(i)
    Load OrderItemPromotionLabel(i)
  Next i

  For i = 1 To MaxSpecialDemandNumber + 4
    Load SPButton(i)
  Next i

  ReadStoreInfo
  ReadSpecialDemand

  For i = 1 To MaxKeyNumber
    Load Key(i)
  Next i


  KeyboardNumber(1, 1) = "Q"
  KeyboardNumber(2, 1) = "W"
  KeyboardNumber(3, 1) = "E"
  KeyboardNumber(4, 1) = "R"
  KeyboardNumber(5, 1) = "T"
  KeyboardNumber(6, 1) = "Y"
  KeyboardNumber(7, 1) = "U"
  KeyboardNumber(8, 1) = "I"
  KeyboardNumber(9, 1) = "O"
  KeyboardNumber(10, 1) = "P"
  KeyboardNumber(1, 2) = "A"
  KeyboardNumber(2, 2) = "S"
  KeyboardNumber(3, 2) = "D"
  KeyboardNumber(4, 2) = "F"
  KeyboardNumber(5, 2) = "G"
  KeyboardNumber(6, 2) = "H"
  KeyboardNumber(7, 2) = "J"
  KeyboardNumber(8, 2) = "K"
  KeyboardNumber(9, 2) = "L"
  KeyboardNumber(1, 3) = "Z"
  KeyboardNumber(2, 3) = "X"
  KeyboardNumber(3, 3) = "C"
  KeyboardNumber(4, 3) = "V"
  KeyboardNumber(5, 3) = "B"
  KeyboardNumber(6, 3) = "N"
  KeyboardNumber(7, 3) = "M"
  KeyboardNumber(8, 3) = ","
  KeyboardNumber(9, 3) = "Enter"
  KeyboardNumber(1, 4) = "Exit"
  KeyboardNumber(2, 4) = "CL"
  KeyboardNumber(3, 4) = "Space"
  KeyboardNumber(4, 4) = "."
  KeyboardNumber(5, 4) = "Backspace"

  ReadTableInfo
  ReadCounterNumber
End Sub

Private Sub Form_Resize()
  ShowLoginFrame True
  ControlFrame.Height = ScaleHeight
  ExitButton.Top = ScaleHeight - ExitButton.Height - 100
  ExitButton.Left = (ControlFrame.Width - ExitButton.Width) / 2

  'Peedy.Left = mOrderForm.CurrentX
  'Peedy.Top = mOrderForm.CurrentY
  'If playAgent Then Peedy.Show
  'Peedy.Speak "\Spd=150\Welcome to use \spd=100\mOrder System \spd=150\Counter \Pit=50\side!"
  'Peedy.Hide
```

```
  For i = 0 To 3
    If i = 0 Then
      Picture1(0).Left = ScaleWidth / 2 - Picture1(0).Width * 1.5
      Picture1(0).Top = (ScaleHeight / 2 - Picture1(0).Height * 1.5 - Picture1(1).Height)
    ElseIf i = 1 Then
      Picture1(i).Left = (ScaleWidth - Picture1(1).Width - Picture1(2).Width - Picture1(3).Width) / 2
    Else
      Picture1(i).Left = Picture1(i - 1).Left + Picture1(i - 1).Width
    End If
    Picture1(i).Top = Picture1(0).Top + Picture1(0).Height
  Next i
  MystoreLabel.Left = Picture1(2).Left 'ScaleWidth - MystoreLabel.Width - Picture1(2).Width / 2
  MystoreLabel.Top = Picture1(1).Top + Picture1(1).Height + margin * 5
End Sub

Private Sub GetOrderNumber()
  sSQLQuery = "INSERT INTO Orders(TableNumber, BeginningDate, BeginningTime," + _
              "WaiterNumber, SpecialDemand, AddTimes, PocketPCIPAddress) " + _
          "VALUES (" + CStr(tableNumber(orderTableNumber)) + ",'" + _
              CStr(Date) + "','" + CStr(Time) + "'," + _
              WaiterNumber + "','" + orderSpecialDemand + "',1,'" + CPCIPAddress + "')"
  oConn.Open dbEngine
  oConn.Execute sSQLQuery

  sSQLQuery = "SELECT MAX(OrderNumber) AS NewOrder FROM Orders WHERE TableNumber=" &
tableNumber(orderTableNumber)
  rs.Open sSQLQuery, oConn
  orderNumber = rs("NewOrder")

  oConn.Close
End Sub



Private Sub HideDiscountFrameButton_Click()
  ShowDiscountFrame False
End Sub

Private Sub HidePaymentFrameButton_Click()
  ShowPaymentFrame False
End Sub

Private Sub HideStatusFrameButton_Click()
  ShowStatusFrame False
End Sub
Private Sub InsertOrderItems2DB()
  addTimes = addTimes + 1
  sSQLQuery = "UPDATE Orders SET AddTimes=" + CStr(addTimes) + _
          " WHERE OrderNumber=" + orderNumber
  oConn.Open dbEngine
  oConn.Execute sSQLQuery
  oConn.Close

  oConn.Open dbEngine
  i = 1
  Do Until OrderItemNameLabel(i).Caption = ""
    sSQLQuery = "INSERT INTO OrderItem(OrderNumber, ItemNumber, Quantity, " + _
            "SpecialDemand, CookingSpecialDemand, AddMoney, Status, AddTimes, Promotion, AddDate, AddTime)
" + _
            "VALUES (" & orderNumber & "," & OrderItemNumberLabel(i).Caption + "," + _
              OrderItemQuantityLabel(i).Caption & ",'" + _
              OrderItemDemandLabel(i).Caption + "','" + _
              PrOrderItemDemandLabel(i).Caption + "'," + _
              OrderItemAddMoneyLabel(i).Caption + ",0," + _
              CStr(addTimes) + "," + OrderItemPromotionLabel(i).Caption + ",'" + _
              CStr(Date) + "','" + CStr(Time) + "')"
```

```
    oConn.Execute sSQLQuery
    i = i + 1
  Loop
  oConn.Close
End Sub


Private Sub ItemButton_Click(Index As Integer)
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Then
  Else
    itemIndex = Index
    For i = 1 To totalItemNumbers(categoryIndex)
      ItemButton(i).BackColor = &H808000
    Next i
    ItemButton(Index).BackColor = &H8080&
    mOrderForm.Caption = "Order Menu"

    ShowNumericalFrame True
  End If
End Sub

Private Sub Key_Click(Index As Integer)
  For i = 1 To MaxKeyNumber
    Key(i).BackColor = ButtonNonpressColor
  Next i
  If keyboardFrameTitle.Caption <> "Input Password" Then Key(Index).BackColor = ButtonPressColor

  If Key(Index).Caption = "Exit" Then
    KeyboardFrameInputLabel.Caption = ""
    keyboardFrameTitle.Caption = ""
    ShowKeyboardFrame False
  ElseIf Key(Index).Caption = "CL" Then
    If actionSign = "Login" Then
      LoginLabel.Caption = ""
    ElseIf actionSign = "Password" Then
      PasswordLabel.Caption = ""
      Password = ""
    Else
      KeyboardFrameInputLabel.Caption = ""
    End If
  ElseIf Key(Index).Caption = "Enter" Then
    If actionSign = "Add Money" Or actionSign = "Cash" Or Mid(actionSign, 1, 8) = "Discount" Then
      If IsNumeric(KeyboardFrameInputLabel.Caption) Then
        If actionSign = "Add Money" Then
          OrderItemAddMoneyLabel(orderItemIndex) = FormatNumber(KeyboardFrameInputLabel.Caption, 2,
vbTrue)
          SPFrameInputLabel.Caption = ""
          keyboardFrameTitle.Caption = ""
          ShowKeyboardFrame False
          SPFrameTitle.Caption = OrderItemNameLabel(orderItemIndex).Caption
          'ShowSPFrame True
        ElseIf actionSign = "Cash" Then
          If CDbl(KeyboardFrameInputLabel.Caption - TotalLabel.Caption) >= 0 Then
            keyboardFrameTitle.Caption = ""
            ShowKeyboardFrame False
            PaymentLabel.Caption = FormatNumber(KeyboardFrameInputLabel.Caption, 2, vbTrue)
            ChangeLabel.Caption = FormatNumber(PaymentLabel.Caption - TotalLabel.Caption, 2, vbTrue)
          Else
            Peedy.Show
            Peedy.Speak "Sorry, the cash is less than total money $" + TotalLabel.Caption + "!"
            Peedy.Hide
          End If
        ElseIf Mid(actionSign, 1, 8) = "Discount" Then
          If CDbl(KeyboardFrameInputLabel.Caption) < 1# And CDbl(KeyboardFrameInputLabel.Caption) > 0 Then
            keyboardFrameTitle.Caption = ""
            ShowKeyboardFrame False
```

```
                discount = CDbl(FormatNumber(KeyboardFrameInputLabel.Caption, 2, vbTrue))
                SubtotalLabel.Caption = FormatNumber(SubtotalLabel.Caption * KeyboardFrameInputLabel.Caption, 2,
vbTrue)
                TaxLabel.Caption = FormatNumber(SubtotalLabel.Caption * taxRate, 2, vbTrue)
                TotalLabel.Caption = FormatNumber(CDbl(SubtotalLabel.Caption) + CDbl(TaxLabel.Caption), 2,
vbTrue)
                PaymentDiscountButton.Enabled = False
                If actionSign = "Discount" Then ShowPaymentFrame True
                WriteOrderDiscount
                If actionSign = "Discount Print" Then PrintBill orderNumber
              Else
                Peedy.Show
                Peedy.Speak "Sorry, the discount must be greater than 0 and less than 1!"
                Peedy.Hide
              End If
            End If
          Else
            Peedy.Show
            Peedy.Speak "Sorry, you must input a number!"
            Peedy.Hide
          End If
        ElseIf actionSign = "Special Demand" And mOrderForm.Caption = "Order List" Then
          OrderItemDemandLabel(orderItemIndex).Caption = SPFrameInputLabel.Caption + " " + _
            KeyboardFrameInputLabel.Caption
          PrOrderItemDemandLabel(orderItemIndex).Caption = prSPFrameInputLabel + " " + _
            KeyboardFrameInputLabel.Caption
          SPFrameInputLabel.Caption = ""
          prSPFrameInputLabel = ""
          keyboardFrameTitle.Caption = ""
          ShowKeyboardFrame False
        ElseIf actionSign = "Login" Then
          If LoginLabel.Caption <> "" Then
            keyboardFrameTitle.Caption = ""
            ShowKeyboardFrame False
          End If
        ElseIf actionSign = "Password" Then
           If Password <> "" Then
            keyboardFrameTitle.Caption = ""
            ShowKeyboardFrame False
            ReadWaiterInfo
          End If
        End If
        KeyboardFrameInputLabel.Caption = ""
      ElseIf Key(Index).Caption = "SH" Then
      ElseIf Key(Index).Caption = "Backspace" Then
        If actionSign = "Login" Then
          length = Len(LoginLabel.Caption)
        ElseIf actionSign = "Password" Then
          length = Len(Password)
        Else
          length = Len(KeyboardFrameInputLabel.Caption)
        End If
        If length > 0 Then
          If actionSign = "Login" Then
            LoginLabel.Caption = Mid(LoginLabel.Caption, 1, length - 1)
          ElseIf actionSign = "Password" Then
            PasswordLabel.Caption = Mid(PasswordLabel.Caption, 1, length - 1)
            Password = Mid(Password, 1, length - 1)
          Else
            KeyboardFrameInputLabel.Caption = Mid(KeyboardFrameInputLabel.Caption, 1, length - 1)
          End If
        End If
      ElseIf Key(Index).Caption = "Space" Then
        If actionSign = "Login" Then
          LoginLabel.Caption = LoginLabel.Caption + " "
        ElseIf actionSign = "Password" Then
          PasswordLabel.Caption = PasswordLabel.Caption + "*"
```

```
        Password = Password + " "
      Else
        KeyboardFrameInputLabel.Caption = KeyboardFrameInputLabel.Caption + " "
      End If
    Else
      If actionSign = "Login" Then
        LoginLabel.Caption = LoginLabel.Caption + Key(Index).Caption
      ElseIf actionSign = "Password" Then
        PasswordLabel.Caption = PasswordLabel.Caption + "*"
        Password = Password + Key(Index).Caption
      Else
        KeyboardFrameInputLabel.Caption = KeyboardFrameInputLabel.Caption + Key(Index).Caption
      End If
    End If
End Sub

Private Sub KitchenSocket_Close()
    Call KitchenSocket.Close
    Call KitchenSocket.Listen
End Sub

Private Sub KitchenSocket_ConnectionRequest(ByVal requestID As Long)
    If KitchenSocket.State <> sckClosed Then
      Call KitchenSocket.Close
    End If

    Call KitchenSocket.Accept(requestID)
End Sub

Private Sub KitchenSocket_DataArrival(ByVal bytesTotal As Long)
    Call KitchenSocket.GetData(rkMessage)
    If Mid(rkMessage, 1, 8) = "\emp\Hi," Then
      Peedy.Show
      Peedy.Speak rkMessage
      Peedy.Hide
    End If
End Sub

Private Sub LanguageButton_Click()
    If Not NumericalFrame.Visible Then
      If LanguageButton.Caption = "Chinese" Then
        LanguageButton.Caption = "English"
        itemLanguage = "ItemName"
      ElseIf LanguageButton.Caption = "English" Then
        LanguageButton.Caption = "Chinese"
        itemLanguage = "ChItemName"
      End If

      languageSwitch = LanguageButton.Caption
      ReadFoodCategory
      ReadItemInfo
      ReadSpecialDemand
      If mOrderForm.Caption = "Sales Report" Then
        If WaiterNumber = Administrator Then
          ShowSalesReportFrame True
        Else
          ShowSampleSalesReportFrame True
        End If
      End If
    End If
End Sub

Private Sub LoginLabel_Click()
    LoginLabel.Caption = ""
    PasswordLabel.Caption = ""
    actionSign = "Login"
    keyboardFrameTitle.Caption = "Input Login Name"
```

```
  ShowKeyboardFrame True
End Sub

Private Sub LogoffButton_Click()
  End
End Sub

Private Sub NighttimeButton_Click()
  If Not NumericalFrame.Visible Then
    totalSales = 0
    startTime = "17:00:00"
    endTime = "23:59:59"
    ReportLabel.Caption = "Night Sales Report"
    ReadAllOrderItems
  End If
End Sub

Private Sub No_Click()
  ExitFrame.Visible = False
End Sub

Private Sub NoDiscountButton_Click()
  ShowDiscountFrame False
  mOrderForm.Caption = CancelButton.Caption
  If Mid(StatusTitleButton.Caption, 1, 5) <> "Table" Then
    TableNoLabel.Caption = tableNumberString
  End If
  ClearOrderItemLabels
  ReadOrderItem2Checkout
  ShowOrderItemFrame True
  PrintBill orderNumber
End Sub

'Private Sub NumericalFrameTitle_Click()
'  NumericalInputLabel.Caption = ""
'  ShowNumericalFrame False
'End Sub

'Private Sub NumericalInputLabel_Click()
'  NumericalInputLabel.Caption = ""
'  ShowNumericalFrame False
'End Sub

Private Sub NumericalKeys_Click(Index As Integer)
  If Index = 10 Then
    NumericalInputLabel.Caption = ""
    ShowNumericalFrame False
  ElseIf Index = 11 Then
    NumericalInputLabel.Caption = ""
  ElseIf Index = 12 Then
    If NumericalInputLabel.Caption <> "" Then
      If CInt(NumericalInputLabel.Caption) <> 0 Then
        If mOrderForm.Caption = "Order Menu" Then
          If totalOrderItemNumber + alreadyOrderItemNumber < MaxOrderItemNumber Then
            totalOrderItemNumber = totalOrderItemNumber + 1
            OrderItemNameLabel(totalOrderItemNumber).Caption = itemNames(itemIndex, categoryIndex)
            OrderItemQuantityLabel(totalOrderItemNumber).Caption = NumericalInputLabel.Caption
            OrderItemDemandLabel(totalOrderItemNumber).Caption = ""
            PrOrderItemDemandLabel(totalOrderItemNumber).Caption = ""
            OrderItemNumberLabel(totalOrderItemNumber).Caption = itemNumbers(itemIndex, categoryIndex)
            OrderItemAddMoneyLabel(totalOrderItemNumber).Caption = "0.00"
            OrderItemPromotionLabel(totalOrderItemNumber).Caption = "0.00"
            readOrderItemNumber = totalOrderItemNumber
            ShowOrderItemFrame True
          End If
          ConfirmButton.Enabled = True
        ElseIf mOrderForm.Caption = "Order List" Then
```

```
            OrderItemQuantityLabel(orderItemIndex).Caption = NumericalInputLabel.Caption
          ElseIf mOrderForm.Caption = "Sales Report" Then
            DeleteAOrder NumericalInputLabel.Caption
          End If
          NumericalInputLabel.Caption = ""
          ShowNumericalFrame False
        Else
          NumericalInputLabel.Caption = ""
          Peedy.Show
          Peedy.Speak "Sorry, you must input a non-zero number!"
          Peedy.Hide
        End If
      Else
        Peedy.Show
        Peedy.Speak "Sorry, you must input a number!"
        Peedy.Hide
      End If
    Else
      NumericalInputLabel.Caption = NumericalInputLabel.Caption + CStr(Index)
    End If
End Sub

Private Sub OrderButton_Click()
  If Not NumericalFrame.Visible Then
    ShowOrderFrame True
    ShowSalesReportFrame False
    ShowEditFrame False
  End If
End Sub

Private Sub OrderItemAddMoneyLabel_Click(Index As Integer)
  If Not NumericalFrame.Visible Then
    orderItemIndex = Index
    UpdateOrderItem
  End If
End Sub

Private Sub OrderItemDemandLabel_Click(Index As Integer)
  If Not NumericalFrame.Visible Then
    orderItemIndex = Index
    UpdateOrderItem
  End If
End Sub

Private Sub OrderItemNameLabel_Click(Index As Integer)
  If Not NumericalFrame.Visible Then
    'MsgBox (mOrderForm.Caption)
    orderItemIndex = Index
    UpdateOrderItem
  End If
End Sub

Private Sub OrderItemQuantityLabel_Click(Index As Integer)
  If Not NumericalFrame.Visible Then
    orderItemIndex = Index
    UpdateOrderItem
  End If
End Sub

Private Sub OrderVScroll_Change()
  OrderItemScrollFrame.Move 0, -OrderVScroll.Value + OrderTitleFrame.Height + margin / 4

End Sub

Private Sub PasswordLabel_Click()
  If LoginLabel.Caption <> "" Then
    PasswordLabel.Caption = ""
```

```
    Password = ""
    actionSign = "Password"
    keyboardFrameTitle = "Input Password"
    ShowKeyboardFrame True
  End If
End Sub

Private Sub Payment()
  sSQLQuery = "UPDATE Orders SET Total=" + FormatNumber(TotalLabel.Caption, , vbTrue, , vbFalse) + ", PayWay
="" + str1 + _
          "", Cashier="" + WaiterNumber + "", SpecialMark=" + CStr(specialMark) + _
          " WHERE OrderNumber =" & orderNumber
  oConn.Open dbEngine
  oConn.Execute sSQLQuery
  oConn.Close
End Sub

Private Sub PaymentATMCardButton_Click()
    ShowPaymentFrame False
    str1 = "ATM"
    PaymentLabel.Caption = TotalLabel.Caption
    ChangeLabel.Caption = "0.00"
End Sub

Private Sub PaymentCashButton_Click()
  str1 = "Cash"
  actionSign = str1
  ShowKeyboardFrame True
  ShowPaymentFrame False
End Sub

Private Sub PaymentCreditCardButton_Click()
  ShowPaymentFrame False
  str1 = "Credit"
  PaymentLabel.Caption = TotalLabel.Caption
  ChangeLabel.Caption = "0.00"
End Sub

Private Sub PaymentDiscountButton_Click()
  actionSign = "Discount"
  ShowPaymentFrame False
  ShowKeyboardFrame True
End Sub

Private Sub PaymentFrameTitle_Click()
  'ShowPaymentFrame False
End Sub

Private Sub PaymentFunction()
  Payment
  TableButtons(orderTableNumber).BackColor = WaitingCheckoutColor
  ShowTableMapFrame True
  '---check out----
  If orderTableNumber <> counterTableNumber Then
    str2 = "True"
    str3 = tableNumber(orderTableNumber)
    str4 = ""
    UpdateTableavailableStatus
    ShowTable True
    Checkout
  End If
End Sub

Private Sub PPCSocket_Close(Index As Integer)
    Call PPCSocket(Index).Close
    Call PPCSocket(Index).Listen
End Sub
```

```
Private Sub PPCSocket_ConnectionRequest(Index As Integer, ByVal requestID As Long)
  If PPCSocket(Index).State <> sckClosed Then
    Call PPCSocket(Index).Close
  End If

  Call PPCSocket(Index).Accept(requestID)
End Sub

Private Sub PPCSocket_DataArrival(Index As Integer, ByVal bytesTotal As Long)
  Call PPCSocket(Index).GetData(rPPCMessage(Index))
  If rPPCMessage(Index) <> "O" Then
    'PrintReceipt rPPCMessage(Index)
  End If
End Sub

Private Sub PrintButton_Click()
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Or SpecialMarkFrame.Visible Then
  Else
    If PrintButton.Caption = "Print Receipt" Then
      PrintReceipt orderNumber
    ElseIf PrintButton.Caption = "Print Order Items" Then
      If orderNumber = "" Then
        addTimes = 0
        GetOrderNumber
        InsertOrderItems2DB
      Else
        InsertOrderItems2DB
      End If
      PrintOrderList orderNumber
      ShowMenuFrame False
      ShowTableMapFrame True
    ElseIf PrintButton.Caption = "Print Order" Then
      PrintOrderList orderNumber
    End If
  End If
End Sub

Private Sub PrintOrderList(orderNo As String)
  oConn.Open dbEngine
  sSQLQuery = "SELECT TableNo, Orders.WaiterNumber AS WN FROM Orders, Tables " + _
        "WHERE OrderNumber=" + orderNo + _
        " AND Orders.TableNumber=Tables.TableNumber"
  rs.Open sSQLQuery, oConn
  tNumber = rs("TableNo")
  wNumber = rs("WN")
  oConn.Close

  If tNumber = counterTableNumber Then
    tNumber = "To Go"
  End If

  oConn.Open dbEngine
  If PrLanguageSwitch = "English" Then
    sSQLQuery = "SELECT ItemName, CookingSpecialDemand, Quantity, AddMoney FROM ORDERITEM, ITEM " +
_
          "WHERE OrderNumber=" & orderNo & " AND OrderItem.AddTimes=" + CStr(addTimes) + _
          " AND ORDERITEM.ItemNumber = ITEM.ItemNumber AND CookingSign=TRUE"
  ElseIf PrLanguageSwitch = "Chinese" Then
    sSQLQuery = "SELECT ChItemName AS ItemName, CookingSpecialDemand, Quantity, AddMoney FROM
ORDERITEM, ITEM " + _
          "WHERE OrderNumber=" & orderNo & " AND OrderItem.AddTimes=" + CStr(addTimes) + _
          " AND ORDERITEM.ItemNumber = ITEM.ItemNumber AND CookingSign=TRUE"
  ElseIf PrLanguageSwitch = "Spanish" Then
    sSQLQuery = "SELECT SpItemName AS ItemName, CookingSpecialDemand, Quantity, AddMoney FROM
ORDERITEM, ITEM " + _
```

```
            "WHERE OrderNumber=" & orderNo & " AND OrderItem.AddTimes=" + CStr(addTimes) + _
            " AND ORDERITEM.ItemNumber = ITEM.ItemNumber AND CookingSign=TRUE"
    End If
    rs.Open sSQLQuery, oConn

    sPrinter.OpenPrinter
    sPrinter.Center
    sPrinter.FontSize (1)
    Print #1, "Order List"; lineFeed;
    sPrinter.FontSize (0)
    Print #1, "Print Time: " + CStr(Now)
    Print #1, "Table No. " + tNumber + "   Waiter: " + wNumber
    sPrinter.LeftJustification

    Do Until rs.EOF
      itemName = rs("ItemName")
      itemQuantity = rs("Quantity")
      Print #1, "NAME: " + FixedLengthString(itemName, 36)
      Print #1, "QUANTITY: " + FixedLengthString(itemQuantity, 4) + _
            "   ADD MONEY:" + FixedLengthString(FormatNumber(rs("AddMoney"), 2, vbTrue), 7)
      If tNumber = "To Go" Then
        Print #1, "DEMAND: " + rs("CookingSpecialDemand") + " To Go" + lineFeed; ""
      Else
        Print #1, "DEMAND: " + rs("CookingSpecialDemand") + lineFeed
      End If
      rs.MoveNext
    Loop
    sPrinter.ClosePrinter
End Sub

Private Sub PrintBill(PPCorderNumber As String)
On Error Resume Next

    oConn.Open dbEngine
    sSQLQuery = "SELECT TableNo, Orders.WaiterNumber AS WN, Discount FROM Orders, Tables " + _
          "WHERE OrderNumber=" + PPCorderNumber + _
          " AND Orders.TableNumber=Tables.TableNumber"
    rs.Open sSQLQuery, oConn
    tNumber = rs("TableNo")
    wNumber = rs("WN")
    oConn.Close

    If tNumber = counterTableNumber Then
      tNumber = "To Go"
    End If

    oConn.Open dbEngine
    If ReceiptLanguageSwitch = "English" Then
      sSQLQuery = "SELECT ItemName, Price, Quantity, AddMoney FROM ORDERITEM, ITEM " + _
            "WHERE OrderNumber=" & PPCorderNumber & " AND ORDERITEM.ItemNumber = ITEM.ItemNumber "
    ElseIf ReceiptLanguageSwitch = "Chinese" Then
      sSQLQuery = "SELECT ChItemName AS ItemName, Price, Quantity, AddMoney FROM ORDERITEM, ITEM " + _
            "WHERE OrderNumber=" & PPCorderNumber & " AND ORDERITEM.ItemNumber = ITEM.ItemNumber "
    ElseIf ReceiptLanguageSwitch = "Spanish" Then
      sSQLQuery = "SELECT SpItemName AS ItemName, Price, Quantity, AddMoney FROM ORDERITEM, ITEM " + _
            "WHERE OrderNumber=" & PPCorderNumber & " AND ORDERITEM.ItemNumber = ITEM.ItemNumber "
    End If
    rs.Open sSQLQuery, oConn

    sPrinter.OpenPrinter
    sPrinter.Center
    sPrinter.FontSize (1)
    Print #1, storeName; lineFeed;
    sPrinter.FontSize (0)
    Print #1, "(" + Mid(phoneNumber, 1, 3) + ")" + Mid(phoneNumber, 4, 7); lineFeed;
    Print #1, Address; lineFeed;
    Print #1, City + ", " + State + zipCode; lineFeed
```

```
  Print #1, "Printing Time: " + CStr(Now)
  Print #1, "Table No. " + tNumber + "   Waiter: " + wNumber
  Print #1, "OrderNumber: " + PPCorderNumber
  sPrinter.LeftJustification

  subtotal = 0
  Do Until rs.EOF
    itemName = rs("ItemName")
    itemQuantity = rs("Quantity")
    addMoney = rs("AddMoney")
    itemProduct = rs("Quantity") * (rs("Price") + addMoney)
    subtotal = subtotal + itemProduct
    Print #1, FixedLengthString(itemName, 28) + _
      " " + FixedLengthString(itemQuantity, 4) + _
      " " + FixedLengthString(FormatNumber(itemProduct, 2, vbTrue), 8)
    rs.MoveNext
  Loop
  If discount <> 0# Then
    subtotal = round(subtotal * discount,2)
  End If
  tax = round(subtotal * taxRate,2)
  total = subtotal + tax
  Print #1, ""
  Print #1, FixedLengthString("SUBTOTAL", 34) + FixedLengthString(FormatNumber(subtotal, 2, vbTrue), 8)
  Print #1, FixedLengthString("TAX (" + FormatNumber(taxRate * 100, 2, vbTrue) + "%)", 34) +
FixedLengthString(FormatNumber(tax, 2, vbTrue), 8)
  Print #1, FixedLengthString("TOTAL", 34) + FixedLengthString("$" + FormatNumber(total, 2, vbTrue), 8)
  sPrinter.ClosePrinter
  oConn.Close
End Sub

Private Sub PrintReceipt(PPCorderNumber As String)
On Error Resume Next

  oConn.Open dbEngine
  sSQLQuery = "SELECT TableNo, Orders.WaiterNumber AS WN, Discount FROM Orders, Tables " + _
        "WHERE OrderNumber=" + PPCorderNumber + _
        " AND Orders.TableNumber=Tables.TableNumber"
  rs.Open sSQLQuery, oConn
  tNumber = rs("TableNo")
  wNumber = rs("WN")
  oConn.Close

  If tNumber = counterTableNumber Then
    tNumber = "To Go"
  End If

  oConn.Open dbEngine
  If ReceiptLanguageSwitch = "English" Then
    sSQLQuery = "SELECT ItemName, Price, Quantity, AddMoney FROM ORDERITEM, ITEM " + _
          "WHERE OrderNumber=" & PPCorderNumber & " AND ORDERITEM.ItemNumber = ITEM.ItemNumber "
  ElseIf ReceiptLanguageSwitch = "Chinese" Then
    sSQLQuery = "SELECT ChItemName AS ItemName, Price, Quantity, AddMoney FROM ORDERITEM, ITEM " + _
          "WHERE OrderNumber=" & PPCorderNumber & " AND ORDERITEM.ItemNumber = ITEM.ItemNumber "
  ElseIf ReceiptLanguageSwitch = "Spanish" Then
    sSQLQuery = "SELECT SpItemName AS ItemName, Price, Quantity, AddMoney FROM ORDERITEM, ITEM " + _
          "WHERE OrderNumber=" & PPCorderNumber & " AND ORDERITEM.ItemNumber = ITEM.ItemNumber "
  End If
  rs.Open sSQLQuery, oConn

  sPrinter.OpenPrinter
  sPrinter.Center
  sPrinter.FontSize (1)
  Print #1, storeName; lineFeed;
  sPrinter.FontSize (0)
  Print #1, "(" + Mid(phoneNumber, 1, 3) + ")" + Mid(phoneNumber, 4, 7); lineFeed;
  Print #1, Address; lineFeed;
```

```
  Print #1, City + ", " + State + zipCode; lineFeed
  Print #1, "Checkout Time: " + CStr(Now)
  Print #1, "Table No. " + tNumber + "   Cashier: " + WaiterNumber
  Print #1, "OrderNumber: " + PPCorderNumber
  sPrinter.LeftJustification

  subtotal = 0
  Do Until rs.EOF
    itemName = rs("ItemName")
    itemQuantity = rs("Quantity")
    addMoney = rs("AddMoney")
    itemProduct = rs("Quantity") * (rs("Price") + addMoney)
    subtotal = subtotal + itemProduct
    Print #1, FixedLengthString(itemName, 28) + _
      " " + FixedLengthString(itemQuantity, 4) + _
      " " + FixedLengthString(FormatNumber(itemProduct, 2, vbTrue), 8)
    rs.MoveNext
  Loop
  If discount <> 0# Then
    subtotal = round(subtotal * discount,2)
  End If
  tax = round(subtotal * taxRate,2)
  total = subtotal + tax
  Print #1, ""
  Print #1, FixedLengthString("SUBTOTAL", 34) + FixedLengthString(FormatNumber(subtotal, 2, vbTrue), 8)
  Print #1, FixedLengthString("TAX (" + FormatNumber(taxRate * 100, 2, vbTrue) + "%)", 34) +
FixedLengthString(FormatNumber(tax, 2, vbTrue), 8)
  Print #1, FixedLengthString("TOTAL", 34) + FixedLengthString("$" + FormatNumber(total, 2, vbTrue), 8)

  Print #1, FixedLengthString("Type of Payment: ", 34) + FixedLengthString(CStr(str1), 8)
  If str1 = "Cash" Then
    Print #1, FixedLengthString("Payment: ", 34) + FixedLengthString(FormatNumber(PaymentLabel.Caption, 2,
vbTrue), 8)
    Print #1, FixedLengthString("Change: ", 34) + FixedLengthString(FormatNumber(ChangeLabel.Caption, 2,
vbTrue), 8)
  End If

  sPrinter.ClosePrinter
  oConn.Close
End Sub

Private Sub ReadAccountingInfo()
  sSQLQuery = "SELECT EndDate, EndTime FROM Accounting WHERE AccountingID=1"
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  If rs("EndDate") <> Date Then
    startTime = "00:00:00"
  Else
    startTime = rs("EndTime")
  End If
  endTime = Time
  oConn.Close
End Sub

Private Sub ReadAllOrderItems()
On Error Resume Next
  timeString = "EndDate=#" + CStr(Date) + "# AND EndTime BETWEEN " + _
        "#" + startTime + "# AND #" + endTime + "#"

  totalSales = 0
  sSQLQuery = "SELECT Sum(Total) AS totalSales FROM Orders WHERE " + timeString
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  totalSales = rs("totalSales")
  oConn.Close
  SaleLabel.Caption = FormatNumber(totalSales, 2, vbTrue)
```

```
  bigSales = 0
  sSQLQuery = "SELECT Sum(Total) AS totalSales FROM Orders WHERE SpecialMark=TRUE AND " + timeString
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  bigSales = rs("totalSales")
  oConn.Close
  SaleLabel.Caption = SaleLabel.Caption + "(B:" + FormatNumber(bigSales, 2, vbTrue)


  smallSales = 0
  sSQLQuery = "SELECT Sum(Total) AS totalSales FROM Orders WHERE SpecialMark=FALSE AND " + timeString
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  smallSales = rs("totalSales")
  oConn.Close
  SaleLabel.Caption = SaleLabel.Caption + " S:" + FormatNumber(smallSales, 2, vbTrue) + ")"

  totalSales = 0
  sSQLQuery = "SELECT Sum(Total) AS totalMoneySales FROM Orders WHERE " + "PayWay='Cash' AND " +
timeString
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  totalSales = rs("totalMoneySales")
  oConn.Close
  CashMoneyLabel.Caption = FormatNumber(totalSales, 2, vbTrue)

  totalSales = 0
  sSQLQuery = "SELECT Sum(Total) AS totalMoneySales FROM Orders WHERE " + "PayWay='Credit' AND " +
timeString
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  totalSales = rs("totalMoneySales")
  oConn.Close
  CreditMoneyLabel.Caption = FormatNumber(totalSales, 2, vbTrue)

  totalSales = 0
  sSQLQuery = "SELECT Sum(Total) AS totalMoneySales FROM Orders WHERE " + "PayWay='ATM' AND " +
timeString
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  totalSales = rs("totalMoneySales")
  oConn.Close
  ATMMoneyLabel.Caption = FormatNumber(totalSales, 2, vbTrue)

  timeString = "AddDate=#" + CStr(Date) + "# AND AddTime BETWEEN " + _
        "#" + startTime + "# AND #" + endTime + "#"
  If languageSwitch = "English" Then
    sSQLQuery = "SELECT ItemName, Quantity, Price, AddMoney FROM OrderItem, Item " + _
          "WHERE OrderItem.ItemNumber=Item.ItemNumber  AND " + timeString
  ElseIf languageSwitch = "Chinese" Then
    sSQLQuery = "SELECT ChItemName AS ItemName, Quantity, Price, AddMoney FROM OrderItem, Item " + _
          "WHERE OrderItem.ItemNumber=Item.ItemNumber  AND " + timeString
  ElseIf languageSwitch = "Spanish" Then
    sSQLQuery = "SELECT SpItemName AS ItemName, Quantity, Price, AddMoney FROM OrderItem, Item " + _
          "WHERE OrderItem.ItemNumber=Item.ItemNumber  AND " + timeString
  End If
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  i = 0
  Do Until rs.EOF
    i = i + 1
    OrderItemNameLabel(i).Caption = rs("ItemName")
    OrderItemDemandLabel(i).Caption = rs("Quantity")
    rs.MoveNext
  Loop
  oConn.Close
  readOrderItemNumber = i
```

```
End Sub

Private Sub ReadAlreadyOrderItemNumber()
  sSQLQuery = "SELECT COUNT(OrderNumber) as TotalNumber FROM OrderItem " + _
        "WHERE OrderNumber=" + orderNumber
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  alreadyOrderItemNumber = rs("TotalNumber")
  oConn.Close

  sSQLQuery = "SELECT AddTimes FROM Orders " + _
        "WHERE OrderNumber=" + orderNumber
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  addTimes = rs("AddTimes")
  oConn.Close
End Sub

Private Sub ReadCounterNumber()
  sSQLQuery = "SELECT TableNumber FROM Tables WHERE SeatNumber='0'"
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  counterNumber = rs("TableNumber")
  oConn.Close

  For i = 1 To totalTableNumber
    If counterNumber = tableNumber(i) Then
      counterTableNumber = tableNo(i)
    End If
  Next i
End Sub

Private Sub ReadFoodCategory()
  sSQLQuery = "SELECT CategoryName, ChCategoryName FROM FOODCATEGORY"
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn

  i = 0
  Do Until rs.EOF
    i = i + 1
    If languageSwitch = "English" Then
      CategoryName(i) = rs("CategoryName")
    ElseIf languageSwitch = "Chinese" Then
      CategoryName(i) = rs("ChCategoryName")
    ElseIf languageSwitch = "Spanish" Then
      CategoryName(i) = rs("SpCategoryName")
    End If
    rs.MoveNext
  Loop
  totalCategoryNumber = i
  oConn.Close
End Sub

Private Sub ReadItemInfo()
  For j = 1 To totalCategoryNumber
    If languageSwitch = "English" Then
      sSQLQuery = "SELECT ItemNumber, ItemName, Price " + _
            "FROM Item, Subfoodcategory " + _
            "WHERE Item.SubcategoryNumber=Subfoodcategory.SubcategoryNumber AND " + _
            "CategoryNumber=" + CStr(j) + " ORDER BY ItemNumber ASC"
    ElseIf languageSwitch = "Chinese" Then
      sSQLQuery = "SELECT ItemNumber, ChItemName AS ItemName, Price " + _
            "FROM Item, Subfoodcategory " + _
            "WHERE Item.SubcategoryNumber=Subfoodcategory.SubcategoryNumber AND " + _
            "CategoryNumber=" + CStr(j) + " ORDER BY ItemNumber ASC"
    ElseIf languageSwitch = "Spanish" Then
      sSQLQuery = "SELECT ItemNumber, SpItemName AS ItemName, Price " + _
```

```
            "FROM Item, Subfoodcategory " + _
            "WHERE Item.SubcategoryNumber=Subfoodcategory.SubcategoryNumber AND " + _
            "CategoryNumber=" + CStr(j) + " ORDER BY ItemNumber ASC"
    End If
    oConn.Open dbEngine
    rs.Open sSQLQuery, oConn

    i = 0
    Do Until rs.EOF
      i = i + 1
      itemNumbers(i, j) = rs("ItemNumber")
      itemNames(i, j) = rs("ItemName")
      itemPrices(i, j) = rs("Price")
      rs.MoveNext
    Loop
    totalItemNumbers(j) = i
    oConn.Close
  Next j
End Sub

Private Sub ReadMonthlyInfo()
  sSQLQuery = "SELECT ItemNumber, AddDate, SUM(Quantity) AS TotalQuantity " + _
        "FROM OrderItem " + _
        "GROUP BY ItemNumber, AddDate " + _
        "ORDER BY AddDate DESC"

  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  oConn.Close
End Sub

Private Sub ReadOrderItemStatus()
  If languageSwitch = "English" Then
    sSQLQuery = "SELECT SeriesNumber, ItemName, Quantity, Status, OrderItem.AddTimes AS oiAddTimes " + _
          "FROM OrderItem, Item " + _
          "WHERE OrderItem.ItemNumber=Item.ItemNumber AND " + _
          "    OrderNumber=" + orderNumber
  ElseIf languageSwitch = "Chinese" Then
    sSQLQuery = "SELECT SeriesNumber, ChItemName AS ItemName, Quantity, Status, OrderItem.AddTimes AS
oiAddTimes " + _
          "FROM OrderItem, Item " + _
          "WHERE OrderItem.ItemNumber=Item.ItemNumber AND " + _
          "    OrderNumber=" + orderNumber
  ElseIf languageSwitch = "Spanish" Then
    sSQLQuery = "SELECT SeriesNumber, SpItemName AS ItemName, Quantity, Status, OrderItem.AddTimes AS
oiAddTimes " + _
          "FROM OrderItem, Item " + _
          "WHERE OrderItem.ItemNumber=Item.ItemNumber AND " + _
          "    OrderNumber=" + orderNumber
  End If
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  i = 0
  Do Until rs.EOF
    i = i + 1
    OrderItemNumberLabel(i).Caption = rs("SeriesNumber")
    OrderItemNameLabel(i).Caption = rs("ItemName")
    OrderItemDemandLabel(i).Caption = rs("Quantity")
    OrderItemQuantityLabel(i).Caption = rs("oiAddTimes")
    Select Case rs("Status")
      Case 0
        OrderItemAddMoneyLabel(i).Caption = "Waiting"
      Case 1
        OrderItemAddMoneyLabel(i).Caption = "Cooking"
      Case 2
        OrderItemAddMoneyLabel(i).Caption = "Ready"
      Case 3
```

```
            OrderItemAddMoneyLabel(i).Caption = "Offer"
        End Select
        rs.MoveNext
    Loop
    readOrderItemNumber = i
    oConn.Close
End Sub

Private Sub ReadOrderItem2Checkout()
    If languageSwitch = "English" Then
        sSQLQuery = "SELECT ItemName, Quantity, Price, AddMoney " + _
                "FROM OrderItem, Item " + _
                "WHERE OrderItem.ItemNumber=Item.ItemNumber AND " + _
                "     OrderNumber=" + orderNumber
    ElseIf languageSwitch = "Chinese" Then
        sSQLQuery = "SELECT ChItemName AS ItemName, Quantity, Price, AddMoney " + _
                "FROM OrderItem, Item " + _
                "WHERE OrderItem.ItemNumber=Item.ItemNumber AND " + _
                "     OrderNumber=" + orderNumber
    ElseIf languageSwitch = "Spanish" Then
        sSQLQuery = "SELECT SpItemName AS ItemName, Quantity, Price, AddMoney " + _
                "FROM OrderItem, Item " + _
                "WHERE OrderItem.ItemNumber=Item.ItemNumber AND " + _
                "     OrderNumber=" + orderNumber
    End If
    oConn.Open dbEngine
    rs.Open sSQLQuery, oConn
    i = 0
    subtotal = 0
    Do Until rs.EOF
        i = i + 1
        OrderItemNameLabel(i).Caption = rs("ItemName")
        OrderItemDemandLabel(i).Caption = rs("Quantity")
        OrderItemQuantityLabel(i).Caption = FormatNumber(rs("Price"), 2, vbTrue)
        OrderItemAddMoneyLabel(i).Caption = FormatNumber(rs("AddMoney"), 2, vbTrue)
        subtotal = subtotal + rs("quantity") * (rs("price") + rs("AddMoney"))
        rs.MoveNext
    Loop
    readOrderItemNumber = i
    oConn.Close
    If discount <> 0 Then
        subtotal = round(subtotal * discount,2)
    End If
    SubtotalLabel.Caption = FormatNumber(subtotal, 2, vbTrue)
    TaxLabel.Caption = FormatNumber(subtotal * taxRate, 2, vbTrue)
    TotalLabel.Caption = FormatNumber(CDbl(SubtotalLabel.Caption) + CDbl(TaxLabel.Caption), 2, vbTrue)
    TaxTitle.Caption = "Tax (" + CStr(taxRate * 100) + "%):"
End Sub

Private Sub ReadOrderPaymentInfo()
On Error Resume Next
    totalCost = ""
    discount = 0#
    sSQLQuery = "SELECT Total, Discount FROM Orders WHERE OrderNumber=" + orderNumber
    oConn.Open dbEngine
    rs.Open sSQLQuery, oConn
    totalCost = rs("Total")
    discount = rs("Discount")
    oConn.Close
End Sub

Private Sub ReadSpecialDemand()
On Error Resume Next
    sSQLQuery = "SELECT SpecialDemand, ChSpecialDemand, SpSpecialDemand FROM SpecialDemand"
    oConn.Open dbEngine
    rs.Open sSQLQuery, oConn
    i = 0
```

```
    Do Until rs.EOF
      i = i + 1
      If languageSwitch = "Chinese" Then
        SPButton(i).Caption = rs("ChSpecialDemand")
      ElseIf languageSwitch = "Spanish" Then
        SPButton(i).Caption = rs("SpSpecialDemand")
      ElseIf languageSwitch = "English" Then
        SPButton(i).Caption = rs("SpecialDemand")
      End If

      If PrLanguageSwitch = "Spanish" Then
        prSpecialDemand(i) = rs("SpSpecialDemand")
      ElseIf PrLanguageSwitch = "Chinese" Then
        prSpecialDemand(i) = rs("ChSpecialDemand")
      ElseIf PrLanguageSwitch = "English" Then
        prSpecialDemand(i) = rs("SpecialDemand")
      End If
      rs.MoveNext
    Loop
    oConn.Close
    SPButton(MaxSpecialDemandNumber + 1).Caption = "ESC"
    SPButton(MaxSpecialDemandNumber + 2).Caption = "Clear"
    SPButton(MaxSpecialDemandNumber + 3).Caption = "Others"
    SPButton(MaxSpecialDemandNumber + 4).Caption = "Enter"
End Sub

Private Sub ReadStoreInfo()
    oConn.Open dbEngine
    sSQLQuery = "SELECT StoreName, PhoneNumber, Address, City, State, ZipCode, TaxRate FROM STORE"
    rs.Open sSQLQuery, oConn
    storeName = rs("StoreName")
    phoneNumber = rs("PhoneNumber")
    Address = rs("Address")
    City = rs("City")
    State = rs("State")
    zipCode = rs("ZipCode")
    taxRate = rs("TaxRate")
    oConn.Close
End Sub

Private Sub ReadTableInfo()
On Error Resume Next

    oConn.Open dbEngine
    sSQLQuery = "SELECT TableNumber, TableNo, AvailableStatus, SeatNumber, XCoordinate, YCoordinate, Width,
Length, Shape FROM Tables WHERE HallNumber=1"
    rs.Open sSQLQuery, oConn
    i = 0
    Do Until rs.EOF
      i = i + 1
      tableNumber(i) = rs("TableNumber")
      tableNo(i) = rs("TableNO")
      tableAvailableStatus(i) = rs("Availablestatus")
      tableSeatNumber(i) = rs("SeatNumber")
      tableXCoordinate(i) = rs("XCoordinate")
      tableYCoordinate(i) = rs("YCoordinate")
      tableWidth(i) = rs("Width")
      tableLength(i) = rs("Length")
      tableShape(i) = rs("Shape")
      rs.MoveNext
    Loop
    oConn.Close
    totalTableNumber = i
    TableCoordinateConvertion
End Sub

Private Sub ReadToGoOrderInfo()
```

```
  sSQLQuery = "SELECT OrderNumber, BeginningDate, BeginningTime, Total, AddTimes " + _
          "FROM Orders " + _
          "WHERE EndDate IS NULL AND TableNumber= (SELECT TableNumber " + _
                                "FROM Tables " + _
                                "WHERE SeatNumber='0')"
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  i = 0
  Do Until rs.EOF
    i = i + 1
    OrderItemNameLabel(i).Caption = rs("BeginningDate")
    OrderItemDemandLabel(i).Caption = rs("BeginningTime")
    OrderItemQuantityLabel(i).Caption = rs("AddTimes")
    OrderItemNumberLabel(i).Caption = rs("OrderNumber")
    If rs("Total") <> "" Then
      OrderItemAddMoneyLabel(i).Caption = "Yes"
    Else
      OrderItemAddMoneyLabel(i).Caption = "No"
    End If
    rs.MoveNext
  Loop
  readOrderItemNumber = i
  oConn.Close
End Sub

Private Sub ReadTop20SalesInfo()
  If ShowTop20Button.Caption = "Quantitative Chart" Then
    If languageSwitch = "Chinese" Then
      sSQLQuery = _
        "SELECT ChItemName, TotalQuantity " + _
        "FROM Item, (SELECT ItemNumber, SUM(Quantity) AS TotalQuantity " + _
              "FROM OrderItem " + _
              "GROUP BY ItemNumber) AS A " + _
        "WHERE A.ItemNumber=Item.ItemNumber " + _
        "ORDER BY TotalQuantity DESC"
    ElseIf languageSwitch = "English" Then
      sSQLQuery = _
        "SELECT ItemName, TotalQuantity " + _
        "FROM Item, (SELECT ItemNumber, SUM(Quantity) AS TotalQuantity " + _
              "FROM OrderItem " + _
              "GROUP BY ItemNumber) AS A " + _
        "WHERE A.ItemNumber=Item.ItemNumber " + _
        "ORDER BY TotalQuantity DESC"
    End If
  Else
    If languageSwitch = "Chinese" Then
      sSQLQuery = _
        "SELECT ChItemName, TotalCost " + _
        "FROM (SELECT ChItemName, SUM(Quantity*(Price+AddMoney)) AS TotalCost " + _
          "FROM Item, (SELECT SeriesNumber, ItemNumber, Quantity, AddMoney " + _
                "FROM OrderItem) AS A " + _
                "WHERE A.ItemNumber=Item.ItemNumber GROUP BY ChItemName) " + _
        "ORDER BY TotalCost DESC"
    ElseIf languageSwitch = "English" Then
      sSQLQuery = _
        "SELECT ItemName, TotalCost " + _
        "FROM (SELECT ItemName, SUM(Quantity*(Price+AddMoney)) AS TotalCost " + _
          "FROM Item, (SELECT SeriesNumber, ItemNumber, Quantity, AddMoney " + _
                "FROM OrderItem) AS A " + _
                "WHERE A.ItemNumber=Item.ItemNumber GROUP BY ItemName) " + _
        "ORDER BY TotalCost DESC"
    End If
  End If

  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  i = 0
```

```
  Do Until rs.EOF
    i = i + 1
    If i <= Top20Sales Then
      If languageSwitch = "Chinese" Then
        Top20SalesArray(i, 1) = rs("ChItemName")
      Else
        Top20SalesArray(i, 1) = rs("ItemName")
      End If
      If ShowTop20Button.Caption = "Monetary Chart" Then
        Top20SalesArray(i, 2) = rs("TotalCost")
      Else
        Top20SalesArray(i, 2) = rs("TotalQuantity")
      End If
    End If
    rs.MoveNext
  Loop
  oConn.Close
  With StatisticalChart
    .ChartData = Top20SalesArray
    .TitleText = "Top 5's " + ShowTop20Button.Caption
    If ShowTop20Button.Caption = "Monetary Chart" Then
      .ColumnLabel = "Dollars"
    Else
      .ColumnLabel = "Dishes"
    End If
  End With
End Sub

Private Sub ReadUncheckoutOrderInfo()
On Error Resume Next
  totalCost = ""
  discount = 0#
  sSQLQuery = "SELECT MAX(OrderNumber) AS IOrderNumber FROM Orders " + _
        "WHERE EndDate IS NULL AND TableNumber=" + CStr(tableNumber(orderTableNumber))
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  Do Until rs.EOF
    orderNumber = rs("IOrderNumber")
    rs.MoveNext
  Loop
  oConn.Close

  totalCost = ""
  sSQLQuery = "SELECT Total, Discount FROM Orders WHERE OrderNumber=" + orderNumber
  oConn.Open dbEngine
  rs.Open sSQLQuery, oConn
  totalCost = rs("Total")
  discount = rs("Discount")
  oConn.Close
End Sub

Private Sub ReadWaiterInfo()
On Error Resume Next
  oConn.Open dbEngine
  sSQLQuery = "SELECT WaiterNumber FROM Waiter " + _
        "WHERE WaiterNumber='" + LoginLabel.Caption + "' AND Password='" + Password + "'"

  rs.Open sSQLQuery, oConn
  WaiterNumber = rs("WaiterNumber")
  If Err.Number <> 0 Then
    Peedy.Show
    Peedy.Speak "Sorry, your \pit=150\login name \pau=100\or password \pit=100\\pau=100\isn't correct, please try it
again!"
    Peedy.Hide
  Else
    ShowLoginFrame False
    LoginLabel.Caption = ""
```

```
    Password = ""
    PasswordLabel.Caption = ""
    keyboardFrameTitle.Caption = ""
    If WaiterNumber <> Administrator Then
      visibleStatus = False
    Else
      visibleStatus = True
    End If
    StorageReportButton.Visible = visibleStatus
    ScheduleButton.Visible = visibleStatus
    EditWaiterButton.Visible = visibleStatus
    EditCategory.Visible = visibleStatus
    EditSubcategory.Visible = visibleStatus
    EditMenuButton.Visible = visibleStatus
    EditTableButton.Visible = visibleStatus
    EditStoreButton.Visible = visibleStatus
    RemoteMonitor.Visible = visibleStatus
  End If
  oConn.Close
End Sub

Private Sub RefreshButton_Click()
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Or SpecialMarkFrame.Visible Then
  Else
    ShowTableMapFrame True
  End If
End Sub

Private Sub ShowCheckoutFrame(showStatus As Boolean)
  If showStatus Then
    CheckoutFrame.Width = OrderItemFrame.Width
    CheckoutFrame.Height = LabelHeight * 4
    Call CheckoutFrame.Move(0, OrderItemFrame.Height - CheckoutFrame.Height - margin)

    SubtotalTitle.Height = LabelHeight * 4 / 5
    SubtotalTitle.Width = OrderNameTitle.Width
    SubtotalTitle.Top = 0
    SubtotalTitle.Left = 0

    TaxTitle.Height = SubtotalTitle.Height
    TaxTitle.Width = SubtotalTitle.Width
    TaxTitle.Top = SubtotalTitle.Top + SubtotalTitle.Height
    TaxTitle.Left = 0

    TotalTitle.Height = SubtotalTitle.Height
    TotalTitle.Width = SubtotalTitle.Width
    TotalTitle.Top = TaxTitle.Top + TaxTitle.Height
    TotalTitle.Left = 0

    SubtotalLabel.Height = SubtotalTitle.Height
    SubtotalLabel.Width = OrderDemandTitle.Width
    SubtotalLabel.Left = CheckoutFrame.Width - SubtotalLabel.Width - margin
    SubtotalLabel.Top = SubtotalTitle.Top

    TaxLabel.Height = SubtotalTitle.Height
    TaxLabel.Width = SubtotalLabel.Width
    TaxLabel.Left = SubtotalLabel.Left
    TaxLabel.Top = TaxTitle.Top

    TotalLabel.Height = SubtotalTitle.Height
    TotalLabel.Width = SubtotalLabel.Width
    TotalLabel.Left = SubtotalLabel.Left
    TotalLabel.Top = TotalTitle.Top

    PaymentTitle.Height = SubtotalTitle.Height
    PaymentTitle.Width = SubtotalTitle.Width
```

```
      PaymentTitle.Left = 0
      PaymentTitle.Top = TotalTitle.Top + TotalTitle.Height

      ChangeTitle.Height = SubtotalTitle.Height
      ChangeTitle.Width = SubtotalTitle.Width
      ChangeTitle.Top = PaymentTitle.Top + PaymentTitle.Height
      ChangeTitle.Left = SubtotalTitle.Left

      PaymentLabel.Height = SubtotalTitle.Height
      PaymentLabel.Width = SubtotalLabel.Width
      PaymentLabel.Left = SubtotalLabel.Left
      PaymentLabel.Top = PaymentTitle.Top

      ChangeLabel.Height = SubtotalTitle.Height
      ChangeLabel.Width = SubtotalLabel.Width
      ChangeLabel.Left = SubtotalLabel.Left
      ChangeLabel.Top = ChangeTitle.Top
   Else
      PaymentLabel.Caption = ""
      ChangeLabel.Caption = ""
   End If
   CheckoutFrame.Visible = showStatus
End Sub

Private Sub ShowDiscountFrame(showStatus As Boolean)
   If showStatus Then
      DiscountTitle.Caption = "Print " + tableNumberString + "'s Bill"
      DiscountFrame.Top = StatusFrame.Top
      DiscountFrame.Left = StatusFrame.Top
   End If
   If discount = 0# Then
      DiscountButton.Enabled = True
   Else
      DiscountButton.Enabled = False
   End If
   DiscountFrame.Visible = showStatus
End Sub

Private Sub ShowEditFrame(showStatus As Boolean)
   If showStatus Then
      EditFrame.Caption = editActionString
      EditFrame.Move ControlFrame.Width + margin / 2, 0, ScaleWidth - ControlFrame.Width - margin / 2, ScaleHeight
      mOrderDataGrid.Move margin, CategoryDataCombo.Top + CategoryDataCombo.Height + margin,
EditFrame.Width - margin * 2
      mOrderDataGrid.Height = EditFrame.Height - mOrderDataGrid.Top - margin
      mOrderDataGrid.AllowAddNew = True
      mOrderDataGrid.AllowDelete = True
      CategoryLabel.Visible = False
      SubcategoryLabel.Visible = False
      CategoryDataCombo.Visible = False
      SubcategoryDataCombo.Visible = False
      CategoryLabel.Move margin, margin * 7
      CategoryDataCombo.Move CategoryLabel.Left + CategoryLabel.Width, CategoryLabel.Top, EditFrame.Width / 2 -
margin * 5 - CategoryLabel.Width
      SubcategoryAdodc.Move CategoryLabel.Left + CategoryLabel.Width + SubcategoryAdodc.Width,
CategoryLabel.Top, EditFrame.Width / 2 - margin * 5 - CategoryLabel.Width
      SubcategoryLabel.Move CategoryDataCombo.Left + CategoryDataCombo.Width + margin * 5,
CategoryLabel.Top
      SubcategoryDataCombo.Move SubcategoryLabel.Left + SubcategoryLabel.Width + margin,
CategoryDataCombo.Top, CategoryDataCombo.Width
      Select Case editActionString
         Case "Edit Waiter"
            WaiterAdodc.Refresh
            Set mOrderDataGrid.DataSource = WaiterAdodc
            mOrderDataGrid.DefColWidth = mOrderDataGrid.Width / (WaiterAdodc.Recordset.Fields.Count + 1)
         Case "Edit Store"
            StoreAdodc.Refresh
```

```
            mOrderDataGrid.AllowAddNew = False
            mOrderDataGrid.AllowDelete = False
            Set mOrderDataGrid.DataSource = StoreAdodc
            mOrderDataGrid.DefColWidth = mOrderDataGrid.Width / (StoreAdodc.Recordset.Fields.Count + 1)
        Case "Edit Table"
            TableAdodc.Refresh
            Set mOrderDataGrid.DataSource = TableAdodc
            mOrderDataGrid.DefColWidth = mOrderDataGrid.Width / (TableAdodc.Recordset.Fields.Count + 1)
        Case "Edit Category"
            CategoryAdodc.Refresh
            Set mOrderDataGrid.DataSource = CategoryAdodc
            mOrderDataGrid.DefColWidth = mOrderDataGrid.Width / (CategoryAdodc.Recordset.Fields.Count + 1)
        Case "Edit Subcategory"
            SubcategoryAdodc.Refresh
            CategoryLabel.Visible = True
            CategoryDataCombo.Visible = True
            SubcategoryAdodc.Visible = True
            SubcategoryAdodc.Top = 350
            SubcategoryAdodc.Left = 8550
            Set mOrderDataGrid.DataSource = SubcategoryAdodc
            mOrderDataGrid.DefColWidth = mOrderDataGrid.Width / (CategoryAdodc.Recordset.Fields.Count + 1)
        Case "Edit Menu"
            CategoryAdodc.Refresh
            SubcategoryAdodc.Refresh
            CategoryLabel.Visible = True
            CategoryDataCombo.Visible = True
            SubcategoryLabel.Visible = True
            SubcategoryDataCombo.Visible = True
            Set mOrderDataGrid.DataSource = ItemAdodc
            mOrderDataGrid.DefColWidth = mOrderDataGrid.Width / (ItemAdodc.Recordset.Fields.Count + 1)
    End Select
    'mOrderDataGrid.Visible = showStatus
  End If
  EditFrame.Visible = showStatus
End Sub

Private Sub ShowItem()
  j = 0
  k = 1
  For i = 1 To MaxItemNumber
    ItemButton(i).Visible = False
  Next i
  For i = 1 To totalItemNumbers(categoryIndex)
    ItemButton(i).Caption = itemNames(i, categoryIndex)
    ItemButton(i).Width = ItemButtonWidth
    ItemButton(i).Height = ItemButtonHeight
    ItemButton(i).Left = j * ItemButtonWidth
    ItemButton(i).Top = CategoryButton(1).Top + k * (ItemButtonHeight)
    ItemButton(i).BackColor = &H808000
    ItemButton(i).Visible = True
    j = j + 1
    If j = totalCategoryNumber Then
      j = 0
      k = k + 1
    End If
  Next i
End Sub

Private Sub ShowKeyboardFrame(showStatus As Boolean)
  If showStatus Then
    KeyboardFrame.Width = Key(0).Width * 10 + margin * 2
    KeyboardFrame.Height = Key(0).Height * 5 + keyboardFrameTitle.Height + margin * 2
    KeyboardFrame.Left = (ScaleWidth - KeyboardFrame.Width) / 2
    KeyboardFrame.Top = ScaleHeight - KeyboardFrame.Height - margin * 10

    keyboardFrameTitle.Width = KeyboardFrame.Width / 2 - margin
    keyboardFrameTitle.Top = margin / 2
```

```
    keyboardFrameTitle.Left = margin * 2 / 3

    KeyboardFrameInputLabel.Width = keyboardFrameTitle.Width
    KeyboardFrameInputLabel.Top = keyboardFrameTitle.Top
    KeyboardFrameInputLabel.Left = keyboardFrameTitle.Left + keyboardFrameTitle.Width - margin / 4

    For i = 1 To 10
      Key(i).Left = margin * 2 / 3 + (i - 1) * Key(0).Width
      Key(i).Top = margin / 2 + keyboardFrameTitle.Height + keyboardFrameTitle.Top
      If i <> 10 Then
        Key(i).Caption = CStr(i)
      Else
        Key(i).Caption = "0"
      End If
      Key(i).Visible = True
    Next i

    For i = 1 To 10
      Key(i + 10).Left = margin * 2 / 3 + (i - 1) * Key(0).Width
      Key(i + 10).Top = margin / 2 + keyboardFrameTitle.Height + keyboardFrameTitle.Top + Key(0).Height
      Key(i + 10).Caption = KeyboardNumber(i, 1)
      Key(i + 10).Visible = True
    Next i

    For i = 1 To 9
      If i = 1 Then Key(i + 20).Width = Key(0).Width * 1.5
      If i = 9 Then Key(i + 20).Width = Key(0).Width * 1.3
      If i <> 1 Then
        Key(i + 20).Left = margin * 2 / 3 + Key(i + 20 - 1).Left + Key(i + 20 - 1).Width
      Else
        Key(i + 20).Left = margin * 2 / 3
      End If
      Key(i + 20).Top = margin / 2 + keyboardFrameTitle.Height + keyboardFrameTitle.Top + Key(0).Height * 2
      Key(i + 20).Caption = KeyboardNumber(i, 2)
      Key(i + 20).Visible = True
    Next i
    For i = 1 To 9
      Key(i + 30).Left = margin * 2 / 3 + (i - 1) * Key(0).Width
      Key(i + 30).Top = margin / 2 + keyboardFrameTitle.Height + keyboardFrameTitle.Top + Key(0).Height * 3
      Key(i + 30).Caption = KeyboardNumber(i, 3)
      Key(i + 30).Visible = True
      If i = 9 Then Key(i + 30).Width = Key(0).Width * 2
    Next i
    For i = 1 To 5
      If i = 1 Then Key(i + 40).Width = Key(0).Width * 1.5
      If i = 3 Then Key(i + 40).Width = Key(0).Width * 3
      If i = 5 Then Key(i + 40).Width = Key(0).Width * 3.5
      If i <> 1 Then
        Key(i + 40).Left = Key(i + 40 - 1).Left + Key(i + 40 - 1).Width
      Else
        Key(i + 40).Left = margin * 2 / 3
      End If
      Key(i + 40).Top = margin / 2 + keyboardFrameTitle.Height + keyboardFrameTitle.Top + Key(0).Height * 4
      Key(i + 40).Caption = KeyboardNumber(i, 4)
      Key(i + 40).Visible = True
    Next i

    If mOrderForm.Caption = "Order List" Then
      keyboardFrameTitle.Caption = OrderItemNameLabel(orderItemIndex).Caption
    End If
  End If
  KeyboardFrame.Visible = showStatus
End Sub

Private Sub ShowLoginFrame(showStatus As Boolean)
  Dim leftMargin
  If showStatus Then
```

```
    LoginFrame.Move 0, 0, ScaleWidth, ScaleHeight
    leftMargin = (LoginFrame.Width - LoginTitle.Width * 3 - margin) / 2 + ControlFrame.Left + margin

    LoginTitle.Move leftMargin, margin * 50
    PasswordTitle.Move leftMargin, LoginTitle.Top + LoginTitle.Height + margin * 5
    LoginLabel.Move leftMargin + LoginTitle.Width + margin, LoginTitle.Top, LoginTitle.Width * 2, LoginTitle.Height
    PasswordLabel.Move leftMargin + LoginTitle.Width + margin, PasswordTitle.Top, LoginTitle.Width * 2,
LoginTitle.Height

    LogoffButton.Left = (ScaleWidth - LogoffButton.Width) / 2
    LogoffButton.Top = ScaleHeight - LogoffButton.Height - margin * 12
  End If
  LoginFrame.Visible = showStatus
End Sub

Private Sub ShowMenuFrame(showStatus As Boolean)
  If showStatus Then
    ExitOrderButton.Caption = "Back to Table Map"
    mOrderForm.Caption = "Order Menu"
    MenuFrame.Height = ScaleHeight - ExitOrderButton.Height - margin * 1
    MenuFrame.Width = ScaleWidth / 2  '*3 / 5
    Call MenuFrame.Move(0, 0)
    ItemButtonWidth = MenuFrame.Width / totalCategoryNumber

    MenuTitleFrame.Width = MenuFrame.Width
    MenuTitleFrame.Height = LabelHeight * 2 + margin / 4 + 50
    MenuTitleFrame.Top = 0
    MenuTitleFrame.Left = 0

    mOrderTitle.Width = MenuTitleFrame.Width
    mOrderTitle.Height = LabelHeight
    mOrderTitle.Top = 50
    mOrderTitle.Left = 0

    MenuTitle.Width = MenuTitleFrame.Width
    MenuTitle.Height = LabelHeight
    MenuTitle.Top = mOrderTitle.Height + mOrderTitle.Top + margin / 4
    MenuTitle.Left = 0

    For i = 1 To totalCategoryNumber
      CategoryButton(i).Caption = CategoryName(i)
      CategoryButton(i).Height = ItemButtonHeight
      CategoryButton(i).Width = ItemButtonWidth
      CategoryButton(i).Top = MenuTitleFrame.Height + margin / 2
      CategoryButton(i).Left = (i - 1) * ItemButtonWidth
      CategoryButton(i).Visible = True
      CategoryButton(i).BackColor = &H808080
    Next i
    CategoryButton(1).BackColor = &H8080& '&HFFFF& '&HC0C0&

    categoryIndex = 1
    ShowItem
  End If
  MenuFrame.Visible = showStatus
End Sub

Private Sub ShowNumericalFrame(showStatus As Boolean)
  If showStatus Then
    If mOrderForm.Caption = "Order Menu" Then
      NumericalFrame.Left = (MenuFrame.Width - NumericalFrame.Width) / 2
      NumericalFrame.Top = (MenuFrame.Height - NumericalFrame.Height) / 2
      NumericalFrameTitle = itemNames(itemIndex, categoryIndex)
    ElseIf mOrderForm.Caption = "Order List" Then
      NumericalFrame.Left = (OrderItemFrame.Width - NumericalFrame.Width) / 2 + MenuFrame.Width
      NumericalFrame.Top = (OrderItemFrame.Height - NumericalFrame.Height) / 2
      NumericalFrameTitle = itemNames(itemIndex, categoryIndex)
    ElseIf mOrderForm.Caption = "Sales Report" Then
```

```
        NumericalFrame.Left = (SalesReportFrame.Width / 2 - NumericalFrame.Width) / 2 + ControlFrame.Width
        NumericalFrame.Top = DeleteAOrderButton.Top - NumericalFrame.Height
        NumericalFrameTitle = "Input Wanted Order Number"
      End If
    End If
    NumericalFrame.Visible = showStatus
End Sub

Private Sub ShowOrderFrame(showStatus As Boolean)
    If showStatus Then
      OrderFrame.Height = ScaleHeight
      OrderFrame.Width = ScaleWidth
      Call PrintButton.Move(ScaleWidth - PrintButton.Width, _
        ScaleHeight - PrintButton.Height - margin / 4)
      Call OrderFrame.Move(0, 0)
      ExitOrderButton.Top = PrintButton.Top
      ExitOrderButton.Left = 0
      ConfirmButton.Top = PrintButton.Top
      ConfirmButton.Left = PrintButton.Left - ConfirmButton.Width
      ConfirmButton.Enabled = False
      PrintButton.Enabled = False
      RefreshButton.Top = PrintButton.Top
      RefreshButton.Left = ExitOrderButton.Width
      TogoButton.Top = PrintButton.Top
      TogoButton.Left = ExitOrderButton.Width + RefreshButton.Width
      ToGoListButton.Top = PrintButton.Top
      ToGoListButton.Width = ScaleWidth - ExitOrderButton.Width - RefreshButton.Width - _
              ConfirmButton.Width - PrintButton.Width - TogoButton.Width
      ToGoListButton.Left = ExitOrderButton.Width + RefreshButton.Width + TogoButton.Width
    End If
    OrderFrame.Visible = showStatus
    ShowTableMapFrame True
End Sub

Private Sub ShowOrderItems()
    j = 0
    For i = 1 To totalOrderItemNumber
      If OrderItemQuantityLabel(i).Caption <> "" Then
        j = j + 1
        orderItemName(j) = OrderItemNameLabel(i).Caption
        orderItemSpecialDemand(j) = OrderItemDemandLabel(i).Caption
        prOrderItemSpecialDemand(j) = PrOrderItemDemandLabel(i).Caption
        orderItemQuantity(j) = OrderItemQuantityLabel(i).Caption
        OrderItemNumber(j) = OrderItemNumberLabel(i).Caption
        orderItemAddMoney(j) = OrderItemAddMoneyLabel(i).Caption
        orderItemPromotion(j) = OrderItemPromotionLabel(i).Caption
      End If

      OrderItemNameLabel(i).Caption = ""
      OrderItemDemandLabel(i).Caption = ""
      PrOrderItemDemandLabel(i).Caption = ""
      OrderItemQuantityLabel(i).Caption = ""
      OrderItemNumberLabel(i).Caption = ""
      OrderItemAddMoneyLabel(i).Caption = ""
      OrderItemPromotionLabel(i).Caption = ""
    Next i

    totalOrderItemNumber = j
    If totalOrderItemNumber <> 0 Then
      For i = 1 To totalOrderItemNumber
        OrderItemNameLabel(i).Caption = orderItemName(i)
        OrderItemDemandLabel(i).Caption = orderItemSpecialDemand(i)
        PrOrderItemDemandLabel(i).Caption = prOrderItemSpecialDemand(i)
        OrderItemQuantityLabel(i).Caption = orderItemQuantity(i)
        OrderItemNumberLabel(i).Caption = OrderItemNumber(i)
        OrderItemAddMoneyLabel(i).Caption = FormatNumber(orderItemAddMoney(i), 2, vbTrue)
        OrderItemPromotionLabel(i).Caption = orderItemPromotion(i)
```

```
    Next i
  Else
    ConfirmButton.Enabled = False
  End If
End Sub

Private Sub ShowOrderItemFrame(showStatus As Boolean)
  If showStatus Then
    'paymentdiscount.Enabled = True
    If mOrderForm.Caption = "Check Out" Or mOrderForm.Caption = "Payment" Or mOrderForm.Caption = "Print Bill"
Then
      ShowCheckoutFrame True
    Else
      ShowCheckoutFrame False
    End If
    If mOrderForm.Caption = "Order List" Or mOrderForm.Caption = "Order Menu" Then
      OrderNameTitle.Caption = "Order Item Name"
      OrderDemandTitle.Caption = "Special Demand"
      OrderQuantityTitle.Caption = "Quantity"
      OrderAddMoneytitle.Caption = "Add Money"
      OrderFrameTitleLabel.Caption = "Order List"
      OrderItemFrame.Width = ScaleWidth - MenuFrame.Width - margin
      OrderVScroll.Height = OrderItemFrame.Height - OrderTitleFrame.Height
      Call OrderItemFrame.Move(MenuFrame.Width + margin, 0)
      PrintButton.Caption = "Print Order Items"
      If readOrderItemNumber = 0 Then
        PrintButton.Enabled = False
        ConfirmButton.Enabled = False
      Else
        PrintButton.Enabled = True
        ConfirmButton.Enabled = True
      End If
      TogoButton.Enabled = False
      RefreshButton.Enabled = False
      ToGoListButton.Enabled = False
      orderItemButtonColor = ButtonNonpressColor
    'ElseIf mOrderForm.Caption = "Check Out" Then
    '  OrderNameTitle.Caption = "Order Item Name"
    '  OrderDemandTitle.Caption = "Quantity"
    '  OrderQuantityTitle.Caption = "Price"
    '  OrderAddMoneytitle.Caption = "Add Money"
    '  OrderFrameTitleLabel.Caption = "Check Out"
    '  OrderItemFrame.Width = ScaleWidth - TableMapFrame.Width - margin
    '  OrderVScroll.Height = OrderItemFrame.Height - _
    '        OrderTitleFrame.Height - CheckoutFrame.Height - margin
    '  Call OrderItemFrame.Move(TableMapFrame.Width + margin, 0)
    '  PrintButton.Caption = "Print Receipt"
    '  PrintButton.Enabled = True
    '  TogoButton.Enabled = True
    '  ConfirmButton.Enabled = True
    '  RefreshButton.Enabled = True
    '  ToGoListButton.Enabled = True
    '  orderItemButtonColor = BackgroundColor
    ElseIf mOrderForm.Caption = "Payment" Or mOrderForm.Caption = "Print Bill" Then
      OrderNameTitle.Caption = "Order Item Name"
      OrderDemandTitle.Caption = "Quantity"
      OrderQuantityTitle.Caption = "Price"
      OrderAddMoneytitle.Caption = "Add Money"
      OrderFrameTitleLabel.Caption = mOrderForm.Caption
      OrderItemFrame.Width = ScaleWidth - TableMapFrame.Width - margin
      OrderVScroll.Height = OrderItemFrame.Height - _
          OrderTitleFrame.Height - CheckoutFrame.Height - margin
      Call OrderItemFrame.Move(TableMapFrame.Width + margin, 0)
      PrintButton.Caption = "Print Receipt"
      TogoButton.Enabled = True
      If mOrderForm.Caption = "Payment" Then
        PrintButton.Enabled = True
```

```
        ConfirmButton.Enabled = True
      Else
        PrintButton.Enabled = False
        ConfirmButton.Enabled = False
      End If
      RefreshButton.Enabled = True
      ToGoListButton.Enabled = True
      orderItemButtonColor = BackgroundColor
    ElseIf mOrderForm.Caption = "Order Status" Then
      OrderNameTitle.Caption = "Order Item Name"
      OrderDemandTitle.Caption = "Quantity"
      OrderQuantityTitle.Caption = "Order Times"
      OrderAddMoneytitle.Caption = "Item Status"
      OrderFrameTitleLabel.Caption = "Order Status"
      OrderItemFrame.Width = ScaleWidth - TableMapFrame.Width - margin
      OrderVScroll.Height = OrderItemFrame.Height - OrderTitleFrame.Height
      Call OrderItemFrame.Move(TableMapFrame.Width + margin, 0)
      'PrintButton.Caption = "Print Items"
      PrintButton.Enabled = False
      TogoButton.Enabled = True
      ConfirmButton.Enabled = False
      RefreshButton.Enabled = True
      ToGoListButton.Enabled = True
      'orderItemButtonColor = BackgroundColor
      orderItemButtonColor = ButtonNonpressColor
    ElseIf mOrderForm.Caption = "Table Map" Or mOrderForm.Caption = "To Go List" Then
      OrderNameTitle.Caption = "Order Date"
      OrderDemandTitle.Caption = "Order Time"
      OrderQuantityTitle.Caption = "Order Times"
      OrderAddMoneytitle.Caption = "Payment"
      OrderFrameTitleLabel.Caption = "To Go List"
      OrderItemFrame.Width = ScaleWidth - TableMapFrame.Width - margin
      OrderVScroll.Height = OrderItemFrame.Height - OrderTitleFrame.Height
      Call OrderItemFrame.Move(TableMapFrame.Width + margin, 0)
      PrintButton.Caption = "Print Items"
      PrintButton.Enabled = False
      TogoButton.Enabled = True
      ConfirmButton.Enabled = False
      RefreshButton.Enabled = True
      ToGoListButton.Enabled = True
      orderItemButtonColor = ButtonNonpressColor
    End If

    OrderItemFrame.Height = ScaleHeight - ExitOrderButton.Height - margin * 1
    OrderTitleFrame.Width = OrderItemFrame.Width
    OrderTitleFrame.Height = LabelHeight * 2 + margin / 4 + 50
    OrderTitleFrame.Top = 0
    OrderTitleFrame.Left = 0
    TableNoLabel.Caption = tableNumberString

    OrderFrameTitleLabel.Top = 50
    OrderFrameTitleLabel.Left = 0
    OrderFrameTitleLabel.Height = LabelHeight
    OrderFrameTitleLabel.Width = OrderItemFrame.Width / 4

    TableNoLabel.Top = OrderFrameTitleLabel.Top
    TableNoLabel.Left = OrderFrameTitleLabel.Width + margin / 2
    TableNoLabel.Height = LabelHeight
    TableNoLabel.Width = OrderItemFrame.Width / 4

    TimeLabel.Caption = " Time: " + CStr(Time) + " " + CStr(Date)
    TimeLabel.Top = OrderFrameTitleLabel.Top
    TimeLabel.Left = TableNoLabel.Left + TableNoLabel.Width + margin / 2
    TimeLabel.Height = LabelHeight
    TimeLabel.Width = OrderItemFrame.Width - TableNoLabel.Width - OrderFrameTitleLabel.Width

'-----------------------------------------------------------------------------
```

```
    OrderNameTitle.Top = TableNoLabel.Top + TableNoLabel.Height + margin / 4
    OrderNameTitle.Left = 0
    OrderNameTitle.Height = LabelHeight
    OrderNameTitle.Width = OrderItemFrame.Width * 2 / 6

    OrderDemandTitle.Top = OrderNameTitle.Top
    OrderDemandTitle.Left = OrderNameTitle.Width + margin / 2
    OrderDemandTitle.Height = LabelHeight
    OrderDemandTitle.Width = OrderItemFrame.Width * 2 / 6

    OrderAddMoneytitle.Top = OrderNameTitle.Top
    OrderAddMoneytitle.Left = OrderNameTitle.Width + OrderDemandTitle.Width + margin
    OrderAddMoneytitle.Height = LabelHeight
    OrderAddMoneytitle.Width = OrderItemFrame.Width / 8

    OrderQuantityTitle.Top = OrderNameTitle.Top
    OrderQuantityTitle.Left = OrderNameTitle.Width + OrderDemandTitle.Width + _
                OrderAddMoneytitle.Width + margin * 1.4
    OrderQuantityTitle.Height = LabelHeight
    OrderQuantityTitle.Width = OrderItemFrame.Width - OrderQuantityTitle.Left + margin / 4


    OrderVScroll.Top = OrderTitleFrame.Height
    OrderVScroll.Left = OrderItemScrollFrame.Width - OrderVScroll.Width
    OrderVScroll.SmallChange = LabelHeight * 2 + margin
    If readOrderItemNumber > 10 And mOrderForm.Caption = "Payment" Then
      OrderVScroll.Visible = True
      maxHeight = (LabelHeight * 2 + margin) * readOrderItemNumber
      OrderVScroll.Max = maxHeight - (LabelHeight * 2 + margin) * 10
      'OrderVScroll.LargeChange = OrderVScroll.Max / (maxHeight / (LabelHeight * 2 + margin) * 10)
    ElseIf readOrderItemNumber > 12 Then
      If mOrderForm.Caption = "Order Menu" Or mOrderForm.Caption = "Order List" Or mOrderForm.Caption = "To
Go List" Or mOrderForm.Caption = "Order Status" Then
        OrderVScroll.Visible = True
        maxHeight = (LabelHeight * 2 + margin) * readOrderItemNumber
        OrderVScroll.Max = maxHeight - (LabelHeight * 2 + margin) * 12
        'OrderVScroll.LargeChange = OrderVScroll.Max / (maxHeight / (LabelHeight * 2 + margin) * 12)
      End If
    Else
      OrderVScroll.Visible = False
      maxHeight = (OrderItemNameLabel(1).Height + margin) * 12
    End If
    OrderVScroll.LargeChange = OrderVScroll.SmallChange

    OrderVScroll.Value = 0
    OrderItemScrollFrame.Height = maxHeight
    OrderItemScrollFrame.Width = OrderItemFrame.Width
    OrderItemScrollFrame.Top = OrderTitleFrame.Height + margin / 4
    OrderItemScrollFrame.Left = 0

    For i = 1 To MaxOrderItemNumber
      OrderItemNameLabel(i).Top = (i - 1) * (LabelHeight * 2 + margin)
      OrderItemNameLabel(i).Left = 0
      OrderItemNameLabel(i).Height = LabelHeight * 2
      OrderItemNameLabel(i).Width = OrderNameTitle.Width
      OrderItemNameLabel(i).BackColor = orderItemButtonColor
      OrderItemNameLabel(i).Visible = True

      OrderItemDemandLabel(i).Top = OrderItemNameLabel(i).Top
      OrderItemDemandLabel(i).Left = OrderDemandTitle.Left - margin / 2
      OrderItemDemandLabel(i).Height = OrderItemNameLabel(i).Height
      OrderItemDemandLabel(i).Width = OrderDemandTitle.Width
      OrderItemDemandLabel(i).BackColor = orderItemButtonColor
      OrderItemDemandLabel(i).Visible = True

      OrderItemAddMoneyLabel(i).Top = OrderItemNameLabel(i).Top
      OrderItemAddMoneyLabel(i).Left = OrderAddMoneytitle.Left - margin
```

```
        OrderItemAddMoneyLabel(i).Height = OrderItemNameLabel(i).Height
        OrderItemAddMoneyLabel(i).Width = OrderAddMoneytitle.Width
        OrderItemAddMoneyLabel(i).BackColor = orderItemButtonColor
        OrderItemAddMoneyLabel(i).Visible = True

        OrderItemQuantityLabel(i).Top = OrderItemNameLabel(i).Top
        OrderItemQuantityLabel(i).Left = OrderQuantityTitle.Left - margin * 1.4
        OrderItemQuantityLabel(i).Height = OrderItemNameLabel(i).Height
        OrderItemQuantityLabel(i).Width = OrderQuantityTitle.Width + _
                OrderVScroll.Visible * OrderVScroll.Width + margin * 1.4
        OrderItemQuantityLabel(i).BackColor = orderItemButtonColor
        OrderItemQuantityLabel(i).Visible = True
      Next i


      If mOrderForm.Caption = "Table Map" Then
        mOrderForm.Caption = "To Go List"
        tableNumberString = "To Go Order"
        ClearOrderItemLabels
        ReadToGoOrderInfo
        ShowOrderItemFrame True
      End If
    End If
    OrderItemFrame.Visible = showStatus
End Sub

Private Sub ShowPaymentFrame(showStatus As Boolean)
    If showStatus Then
      PaymentFrame.Left = StatusFrame.Top
      PaymentFrame.Top = StatusFrame.Top
    End If
    If discount = 0# Then
      PaymentDiscountButton.Enabled = True
    Else
      PaymentDiscountButton.Enabled = False
    End If
    PaymentFrameTitle.Caption = "Type of Payment of " + tableNumberString
    PaymentFrame.Visible = showStatus
End Sub

Private Sub ShowSalesReportFrame(showStatus As Boolean)

    If showStatus Then
      mOrderForm.Caption = "Sales Report"
      SaleLabel.Caption = ""
      SalesReportFrame.Move ControlFrame.Width + margin / 2, 0, ScaleWidth - ControlFrame.Width - margin / 2,
ScaleHeight

      DaytimeButton.Top = ScaleHeight - DaytimeButton.Height - margin * 2
      WholeDayButton.Top = DaytimeButton.Top
      DeleteAOrderButton.Top = DaytimeButton.Top
      ShowTop20Button.Top = DaytimeButton.Top
      MonthlyChartButton.Top = DaytimeButton.Top
      MarkReportButton.Top = DaytimeButton.Top

      If WaiterNumber = Administrator Then
        DaytimeButton.Left = (SalesReportFrame.Width / 2 - DaytimeButton.Width * 6 - margin * 4) / 2
        WholeDayButton.Left = DaytimeButton.Left + DaytimeButton.Width + margin
        ShowTop20Button.Left = WholeDayButton.Left + WholeDayButton.Width + margin
        MonthlyChartButton.Left = ShowTop20Button.Left + ShowTop20Button.Width + margin
        MarkReportButton.Left = MonthlyChartButton.Left + MonthlyChartButton.Width + margin
        DeleteAOrderButton.Left = MarkReportButton.Left + MarkReportButton.Width + margin
        DeleteAOrderButton.Visible = True
        MarkReportButton.Visible = False
        ShowTop20Button.Visible = True
        MonthlyChartButton.Visible = False
      Else
```

```
        DaytimeButton.Left = (SalesReportFrame.Width / 2 - DaytimeButton.Width * 4 - margin * 3) / 2
        WholeDayButton.Left = DaytimeButton.Left + DaytimeButton.Width + margin
        ShowTop20Button.Left = WholeDayButton.Left + WholeDayButton.Width + margin
        MonthlyChartButton.Left = ShowTop20Button.Left + ShowTop20Button.Width + margin
        DeleteAOrderButton.Visible = False
        MarkReportButton.Visible = False
    End If

    TotalSalesLabel.Top = DaytimeButton.Top - TotalSalesLabel.Height - margin / 2
    TotalSalesLabel.Left = margin
    SaleLabel.Top = TotalSalesLabel.Top
    SaleLabel.Left = TotalSalesLabel.Width + margin
    CashLabel.Top = TotalSalesLabel.Top
    CashLabel.Left = margin
    CashMoneyLabel.Top = TotalSalesLabel.Top
    CashMoneyLabel.Left = CashLabel.Left + CashLabel.Width + margin
    CreditCardLabel.Top = TotalSalesLabel.Top
    CreditCardLabel.Left = CashMoneyLabel.Left + CashMoneyLabel.Width + margin
    CreditMoneyLabel.Top = TotalSalesLabel.Top
    CreditMoneyLabel.Left = CreditCardLabel.Left + CreditCardLabel.Width + margin
    ATMLabel.Top = TotalSalesLabel.Top
    ATMLabel.Left = CreditMoneyLabel.Left + CreditMoneyLabel.Width + margin
    ATMMoneyLabel.Top = TotalSalesLabel.Top
    ATMMoneyLabel.Left = ATMLabel.Left + ATMLabel.Width + margin
    ReportLabel.Top = TotalSalesLabel.Top - ReportLabel.Height - margin / 2
    ReportLabel.Left = TotalSalesLabel.Left

    TotalSalesLabel.Top = ReportLabel.Top
    TotalSalesLabel.Left = ReportLabel.Left + ReportLabel.Width
    SaleLabel.Top = ReportLabel.Top
    SaleLabel.Left = TotalSalesLabel.Left + TotalSalesLabel.Width + margin

    If languageSwitch = "Chinese" Then
        DeletedOrderItemAdodc.RecordSource = _
            "SELECT ChItemName, Quantity, AddDate+AddTime AS AddedTime, " + _
                "DeletedDate+DeletedTime AS DeletedTime, WaiterNumber " + _
            "FROM DeletedOrderItem, Item " + _
            "WHERE Item.ItemNumber=DeletedOrderItem.ItemNumber " + _
            "ORDER BY DeletedDate DESC"
    ElseIf languageSwitch = "English" Then
        DeletedOrderItemAdodc.RecordSource = _
            "SELECT ItemName, Quantity, AddDate+AddTime, " + _
                "DeletedDate+DeletedTime, WaiterNumber " + _
            "FROM DeletedOrderItem, Item " + _
            "WHERE Item.ItemNumber=DeletedOrderItem.ItemNumber " + _
            "ORDER BY DeletedDate DESC"
    End If
    DeletedOrderItemAdodc.Refresh
    Set DeletedOrderItemDataGrid.DataSource = DeletedOrderItemAdodc
    DeletedOrderItemDataGrid.Height = SalesReportFrame.Height / 6
    DeletedOrderItemDataGrid.Move margin, ReportLabel.Top - DeletedOrderItemDataGrid.Height,
SalesReportFrame.Width / 2 - margin

    With OrderListAdodc
        .ConnectionString = "Provider=MSDataShape;Data Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=c:\morder\mOrderDatabase.mdb;"
        .RecordSource = "SHAPE {SELECT OrderNumber, Total, " + _
                            "EndDate,EndTime, Discount, SpecialMark " + _
                        "FROM Orders " + _
                        "ORDER BY OrderNumber DESC} AS Orders " + _
                    "APPEND({SELECT OrderNumber, " + itemLanguage + ", Quantity, " + _
                            "SpecialDemand, AddMoney " + _
                    "FROM OrderItem, Item " + _
                    "WHERE Item.ItemNumber=OrderItem.ItemNumber} " + _
                    "AS OrderItem RELATE " + _
                    "OrderNumber TO OrderNumber) AS Orders"
    End With
```

```
    OrderListAdodc.Refresh
    Set OrderListGrid.DataSource = OrderListAdodc
    OrderListGrid.Width = SalesReportFrame.Width / 2 - margin * 2
    OrderListGrid.Height = SalesReportFrame.Height / 2 - margin * 11
    OrderListGrid.Move SalesReportFrame.Width / 2 + margin, SalesReportFrame.Height - OrderListGrid.Height

    If languageSwitch = "Chinese" Then
      QuantitativeStatisticsAdodc.RecordSource = _
        "SELECT ChItemName, TotalQuantity " + _
        "FROM Item, (SELECT ItemNumber, SUM(Quantity) AS TotalQuantity " + _
              "FROM OrderItem " + _
              "GROUP BY ItemNumber) AS A " + _
        "WHERE A.ItemNumber=Item.ItemNumber " + _
        "ORDER BY TotalQuantity DESC"
    ElseIf languageSwitch = "English" Then
      QuantitativeStatisticsAdodc.RecordSource = _
        "SELECT ItemName, TotalQuantity " + _
        "FROM Item, (SELECT ItemNumber, SUM(Quantity) AS TotalQuantity " + _
              "FROM OrderItem " + _
              "GROUP BY ItemNumber) AS A " + _
        "WHERE A.ItemNumber=Item.ItemNumber " + _
        "ORDER BY TotalQuantity DESC"
    End If
    QuantitativeStatisticsAdodc.Refresh
    Set QuantitativeStatisticalGrid.DataSource = QuantitativeStatisticsAdodc
    QuantitativeStatisticalGrid.Width = DeletedOrderItemDataGrid.Width / 2
    QuantitativeStatisticalGrid.Height = SalesReportFrame.Height / 6
    QuantitativeStatisticalGrid.Move DeletedOrderItemDataGrid.Left, DeletedOrderItemDataGrid.Top -
DeletedOrderItemDataGrid.Height

    If languageSwitch = "Chinese" Then
      MonetaryStatisticsAdodc.RecordSource = _
        "SELECT ChItemName, TotalCost " + _
        "FROM (SELECT ChItemName, SUM(Quantity*(Price+AddMoney)) AS TotalCost " + _
          "FROM Item, (SELECT SeriesNumber, ItemNumber, Quantity, AddMoney " + _
                "FROM OrderItem) AS A " + _
                "WHERE A.ItemNumber=Item.ItemNumber GROUP BY ChItemName) " + _
        "ORDER BY TotalCost DESC"
    ElseIf languageSwitch = "English" Then
      MonetaryStatisticsAdodc.RecordSource = _
        "SELECT ItemName, TotalCost " + _
        "FROM (SELECT ItemName, SUM(Quantity*(Price+AddMoney)) AS TotalCost " + _
          "FROM Item, (SELECT SeriesNumber, ItemNumber, Quantity, AddMoney " + _
                "FROM OrderItem) AS A " + _
                "WHERE A.ItemNumber=Item.ItemNumber GROUP BY ItemName) " + _
        "ORDER BY TotalCost DESC"
    End If
    MonetaryStatisticsAdodc.Refresh
    Set MonetaryStatisticalGrid.DataSource = MonetaryStatisticsAdodc
    MonetaryStatisticalGrid.Width = DeletedOrderItemDataGrid.Width / 2
    MonetaryStatisticalGrid.Height = QuantitativeStatisticalGrid.Height
    MonetaryStatisticalGrid.Move DeletedOrderItemDataGrid.Left + _
      QuantitativeStatisticalGrid.Width, DeletedOrderItemDataGrid.Top - _
      DeletedOrderItemDataGrid.Height

    StatisticalChart.Width = SalesReportFrame.Width - margin
    StatisticalChart.Height = (SalesReportFrame.Width / 5 + margin * 8) * 2
    StatisticalChart.Move margin, 0
    ReadTop20SalesInfo
    If ShowTop20Button.Caption = "Monetary Chart" Then
      ShowTop20Button.Caption = "Quantitative Chart"
    Else
      ShowTop20Button.Caption = "Monetary Chart"
    End If
  Else
    SaleLabel.Caption = ""
```

```
    CashMoneyLabel.Caption = ""
    CreditMoneyLabel.Caption = ""
    ATMMoneyLabel.Caption = ""
  End If
  SalesReportFrame.Visible = showStatus
End Sub


Private Sub ShowSampleSalesReportFrame(showStatus As Boolean)

  If showStatus Then
    mOrderForm.Caption = "Sales Report"
    SaleLabel.Caption = ""
    SalesReportFrame.Move ControlFrame.Width + margin / 2, 0, ScaleWidth - ControlFrame.Width - margin / 2,
ScaleHeight

    DaytimeButton.Top = ScaleHeight - DaytimeButton.Height - margin * 2
    WholeDayButton.Top = DaytimeButton.Top
    DeleteAOrderButton.Top = DaytimeButton.Top
    ShowTop20Button.Top = DaytimeButton.Top
    MonthlyChartButton.Top = DaytimeButton.Top
    MarkReportButton.Top = DaytimeButton.Top

    If WaiterNumber = Administrator Then
      DaytimeButton.Left = (SalesReportFrame.Width / 2 - DaytimeButton.Width * 6 - margin * 4) / 2
      WholeDayButton.Left = DaytimeButton.Left + DaytimeButton.Width + margin
      ShowTop20Button.Left = WholeDayButton.Left + WholeDayButton.Width + margin
      MonthlyChartButton.Left = ShowTop20Button.Left + ShowTop20Button.Width + margin
      MarkReportButton.Left = MonthlyChartButton.Left + MonthlyChartButton.Width + margin
      DeleteAOrderButton.Left = MarkReportButton.Left + MarkReportButton.Width + margin
      DeleteAOrderButton.Visible = True
      MarkReportButton.Visible = False
    Else
      DaytimeButton.Left = (SalesReportFrame.Width / 2 - DaytimeButton.Width * 2 - margin) / 2
      WholeDayButton.Left = DaytimeButton.Left + DaytimeButton.Width + margin
      ShowTop20Button.Left = WholeDayButton.Left + WholeDayButton.Width + margin
      MonthlyChartButton.Left = ShowTop20Button.Left + ShowTop20Button.Width + margin
      ShowTop20Button.Visible = False
      MonthlyChartButton.Visible = False
      DeleteAOrderButton.Visible = False
      MarkReportButton.Visible = False
    End If

    TotalSalesLabel.Top = DaytimeButton.Top - TotalSalesLabel.Height - margin / 2
    TotalSalesLabel.Left = margin
    SaleLabel.Top = TotalSalesLabel.Top
    SaleLabel.Left = TotalSalesLabel.Width + margin
    CashLabel.Top = TotalSalesLabel.Top
'    CashLabel.Left = SaleLabel.Left + SaleLabel.Width + margin
    CashLabel.Left = margin
    CashMoneyLabel.Top = TotalSalesLabel.Top
    CashMoneyLabel.Left = CashLabel.Left + CashLabel.Width + margin
    CreditCardLabel.Top = TotalSalesLabel.Top
    CreditCardLabel.Left = CashMoneyLabel.Left + CashMoneyLabel.Width + margin
    CreditMoneyLabel.Top = TotalSalesLabel.Top
    CreditMoneyLabel.Left = CreditCardLabel.Left + CreditCardLabel.Width + margin
    ATMLabel.Top = TotalSalesLabel.Top
    ATMLabel.Left = CreditMoneyLabel.Left + CreditMoneyLabel.Width + margin
    ATMMoneyLabel.Top = TotalSalesLabel.Top
    ATMMoneyLabel.Left = ATMLabel.Left + ATMLabel.Width + margin
    ReportLabel.Top = TotalSalesLabel.Top - ReportLabel.Height - margin / 2
    ReportLabel.Left = TotalSalesLabel.Left

    TotalSalesLabel.Top = ReportLabel.Top
    TotalSalesLabel.Left = ReportLabel.Left + ReportLabel.Width
    SaleLabel.Top = ReportLabel.Top
    SaleLabel.Left = TotalSalesLabel.Left + TotalSalesLabel.Width + margin
```

```
    With OrderListAdodc
      .ConnectionString = "Provider=MSDataShape;Data Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=c:\morder\mOrderDatabase.mdb;"
      .RecordSource = "SHAPE {SELECT OrderNumber, Total, Payway, TableNumber, WaiterNumber, " + _
                        "EndDate, EndTime, Discount, SpecialMark " + _
                    "FROM Orders " + _
                    "WHERE EndDate=#" + CStr(Date) + "# " + _
                    "ORDER BY OrderNumber DESC} AS Orders " + _
                "APPEND({SELECT OrderNumber, " + itemLanguage + ", Quantity, " + _
                        "AddMoney " + _
                "FROM OrderItem, Item " + _
                "WHERE Item.ItemNumber=OrderItem.ItemNumber} " + _
                "AS OrderItem RELATE " + _
                "OrderNumber TO OrderNumber) AS Orders"
    End With

    OrderListAdodc.Refresh
    Set OrderListGrid.DataSource = OrderListAdodc
    With OrderListGrid
      .Width = SalesReportFrame.Width - margin
      .Height = ReportLabel.Top - margin
      .Move 0, margin
    End With
    OrderListGrid.Refresh
  Else
    SaleLabel.Caption = ""
    CashMoneyLabel.Caption = ""
    CreditMoneyLabel.Caption = ""
    ATMMoneyLabel.Caption = ""
  End If
  SalesReportFrame.Visible = showStatus
End Sub

Private Sub ShowSPFrame(showStatus As Boolean)
  If showStatus Then
    SPFrame.Width = SPButton(0).Width * 6 + margin * 2
    SPFrame.Height = SPButton(0).Height * (MaxSpecialDemandNumber + 4) / 6 + SPFrameTitle.Height + margin *
2
    SPFrame.Top = (ScaleHeight - SPFrame.Height) / 2
    SPFrame.Left = (OrderItemFrame.Width - SPFrame.Width) / 2 + OrderItemFrame.Left

    SPFrameTitle.Caption = OrderItemNameLabel(orderItemIndex).Caption
    SPFrameTitle.Width = SPFrame.Width / 2 - margin * 2
    SPFrameInputLabel.Width = SPFrame.Width / 2
    SPFrameTitle.Top = margin / 2
    SPFrameTitle.Left = margin / 2
    SPFrameInputLabel.Top = SPFrameTitle.Top
    SPFrameInputLabel.Left = SPFrameTitle.Width + SPFrameTitle.Left
    SPFrameInputLabel.Caption = ""
    j = 0
    k = 0
    For i = 1 To MaxSpecialDemandNumber + 4
      SPButton(i).Top = SPFrameTitle.Top + SPFrameTitle.Height + margin / 2 + k * SPButton(0).Height
      SPButton(i).Left = SPFrameTitle.Left + j * SPButton(0).Width
      SPButton(i).Visible = True
      j = j + 1
      If j = 6 Then
        j = 0
        k = k + 1
      End If
    Next i
  End If
  SPFrame.Visible = showStatus
End Sub
```

```
Private Sub ShowSpecialMarkFrame(showStatus As Boolean)
  If showStatus Then
    With SpecialMarkFrame
      .Left = (OrderFrame.Width - .Width) / 2
      .Top = (OrderFrame.Height - .Height) / 2
    End With
  End If
  SpecialMarkFrame.Visible = showStatus
End Sub

Private Sub ShowStatusFrame(showStatus As Boolean)
  If showStatus Then
    If mOrderForm.Caption = "Table Map" Or mOrderForm.Caption = "To Go List" Then
      StatusTitleButton.Caption = tableNumberString
      If Mid(StatusTitleButton.Caption, 1, 5) = "Table" Then
        StatusFrame.Left = (TableMapFrame.Width - StatusFrame.Width) / 2
      Else
        StatusFrame.Left = TableMapFrame.Width + (OrderItemFrame.Width - StatusFrame.Width) / 2
      End If
      StatusFrame.Top = (TableMapFrame.Height - StatusFrame.Height) / 2
      'If orderNumber = "" Then
       ' AddButton.Caption = "Add Item"
       ' CancelButton.Caption = "Cancel Hold"
       ' DemandButton.Caption = "Special Demand"
       ' AddMoneyButton.Caption = "Payment"
       ' AddButton.Enabled = True
       ' DemandButton.Enabled = False
       ' AddMoneyButton.Enabled = False
      'Else
       ' AddButton.Caption = "Add Item"
       ' CancelButton.Caption = "Print Bill"
       ' DemandButton.Caption = "Order Item Status"
       ' AddButton.Enabled = True
       ' DemandButton.Enabled = True
       ' CancelButton.Enabled = True
       ' AddMoneyButton.Enabled = True
      'End If
      '-----------------------------------------------------------
      AddButton.Caption = "Add Item"
      AddMoneyButton.Caption = "Payment"
      AddMoneyButton.Enabled = True
      If orderNumber = "" Then
        CancelButton.Caption = "Cancel Hold"
        DemandButton.Caption = "Special Demand"
        addTimes = 0
      Else
        CancelButton.Caption = "Check Out"
        DemandButton.Caption = "Order Item Status"
        itemAddTimes = "1"
      End If
      If totalCost = "" Then
        AddButton.Enabled = True
        AddMoneyButton.Enabled = True
        CancelButton.Enabled = False
        DemandButton.Enabled = True

      Else
        AddButton.Enabled = False
        AddMoneyButton.Enabled = False
        CancelButton.Enabled = True
        If mOrderForm.Caption = "To Go List" Then
          DemandButton.Enabled = True
        Else
          DemandButton.Enabled = False
        End If
      End If
      If orderNumber = "" Then
```

```
            AddButton.Enabled = True
            AddMoneyButton.Enabled = False
            CancelButton.Enabled = True
            DemandButton.Enabled = False
        End If
        If totalCost = "" And orderNumber <> "" Then
            CancelButton.Enabled = True
            CancelButton.Caption = "Print Bill"
        End If
    ElseIf mOrderForm.Caption = "Order List" Then
        StatusFrame.Top = (OrderItemFrame.Height - StatusFrame.Height) / 2
        StatusFrame.Left = (OrderItemFrame.Width - StatusFrame.Width) / 2 + OrderItemFrame.Left
        StatusTitleButton.Caption = choiceItemName
        AddButton.Caption = "Modify Quantity"
        CancelButton.Caption = "Cancel Item"
        DemandButton.Caption = "Special Demand"
        AddMoneyButton.Caption = "Add Money"
        AddButton.Enabled = True
        AddMoneyButton.Enabled = True
        CancelButton.Enabled = True
        DemandButton.Enabled = True
    ElseIf mOrderForm.Caption = "Order Status" Then
        StatusFrame.Top = (OrderItemFrame.Height - StatusFrame.Height) / 2
        StatusFrame.Left = (OrderItemFrame.Width - StatusFrame.Width) / 2 + OrderItemFrame.Left
        StatusTitleButton.Caption = choiceItemName
        AddButton.Caption = "Modify Status"
        CancelButton.Caption = "Cancel Ordered Item"
        DemandButton.Caption = "Special Demand"
        AddMoneyButton.Caption = "Add Money"
        AddButton.Enabled = False
        AddMoneyButton.Enabled = False
        CancelButton.Enabled = True
        DemandButton.Enabled = False
    End If
  End If
  StatusFrame.Visible = showStatus
End Sub

Private Sub ShowTable(showStatus As Boolean)
  If showStatus Then
    ReadTableInfo
    For i = 1 To totalTableNumber
        TableButtons(i).Left = tableXCoordinate(i)
        TableButtons(i).Top = tableYCoordinate(i)
        TableButtons(i).Width = tableWidth(i)
        TableButtons(i).Height = tableLength(i)
        If tableNo(i) < 10 Then
            tableNumberString = "0" + CStr(tableNo(i))
        Else
            tableNumberString = CStr(tableNo(i))
        End If
        If tableSeatNumber(i) = 0 Then
            tableNumberString = "Counter"
            TableButtons(i).FontSize = 14
        End If
        TableButtons(i).Caption = tableNumberString
        If tableAvailableStatus(i) Then
            TableButtons(i).BackColor = TableAvailableColor
        Else
            TableButtons(i).BackColor = TableUnavailableColor
        End If
        TableButtons(i).Visible = True
    Next i
  Else
    For i = 1 To totalTableNumber
        TableButtons(i).Visible = False
    Next i
```

```
  End If
End Sub

Private Sub ShowTableMapFrame(showStatus As Boolean)

  TableMapFrame.Visible = showStatus
  If showStatus Then
    ExitOrderButton.Caption = "Back to Main Menu"
    mOrderForm.Caption = "Table Map"
    TableMapFrame.Height = ScaleHeight - ExitOrderButton.Height - margin
    TableMapFrame.Width = ScaleWidth / 2
    Call TableMapFrame.Move(0, 0)
    ShowTable showStatus
    tableNumberString = "To Go Order"
    ClearOrderItemLabels
    ShowOrderItemFrame showStatus
    TogoButton.Enabled = True
  End If
End Sub


Private Sub SalesReportButton_Click()
  If Not NumericalFrame.Visible Then
    ShowOrderFrame False
    If WaiterNumber = Administrator Then
      ShowSalesReportFrame True
    Else
      ShowSampleSalesReportFrame True
    End If
    ShowEditFrame False
  End If
End Sub

Private Sub ScheduleButton_Click()
  If Not NumericalFrame.Visible Then
  End If
End Sub

Private Sub ShowTop20Button_Click()
  If Not NumericalFrame.Visible Then
    ReadTop20SalesInfo
    If ShowTop20Button.Caption = "Monetary Chart" Then
      ShowTop20Button.Caption = "Quantitative Chart"
    Else
      ShowTop20Button.Caption = "Monetary Chart"
    End If
  End If
End Sub

Private Sub SMNOButton_Click()
  specialMark = False
  ShowSpecialMarkFrame False
  PaymentFunction
End Sub

Private Sub SMYESButton_Click()
  specialMark = True
  ShowSpecialMarkFrame False
  PaymentFunction
End Sub

Private Sub SPButton_Click(Index As Integer)
  If Index = MaxSpecialDemandNumber + 1 Then
    ShowSPFrame False
  ElseIf Index = MaxSpecialDemandNumber + 2 Then
    SPFrameInputLabel.Caption = ""
```

```
    prSPFrameInputLabel = ""
  ElseIf Index = MaxSpecialDemandNumber + 3 Then
    actionSign = DemandButton.Caption
    ShowKeyboardFrame True
    ShowSPFrame False
  ElseIf Index = MaxSpecialDemandNumber + 4 Then
    OrderItemDemandLabel(orderItemIndex).Caption = SPFrameInputLabel.Caption
    PrOrderItemDemandLabel(orderItemIndex).Caption = prSPFrameInputLabel
    SPFrameInputLabel.Caption = ""
    prSPFrameInputLabel = ""
    ShowSPFrame False
  Else
    If SPFrameInputLabel.Caption = "" Then
      SPFrameInputLabel.Caption = SPButton(Index).Caption
      prSPFrameInputLabel = prSpecialDemand(Index)
    Else
      SPFrameInputLabel.Caption = SPFrameInputLabel.Caption + ", " + SPButton(Index).Caption
      prSPFrameInputLabel = prSPFrameInputLabel + ", " + prSpecialDemand(Index)
    End If
  End If
End Sub

Private Sub SubmitButton_Click()
  ShowLoginFrame False
End Sub

Private Sub StatusTitleButton_Click()
  'ShowStatusFrame False
End Sub

Private Sub StorageReportButton_Click()
  If WaiterNumber = Administrator And Not NumericalFrame.Visible Then
    editActionString = "Storage Report"
    ShowEditFrame False
    ShowOrderFrame False
    ShowSalesReportFrame False
  End If
End Sub

Private Sub SubcategoryDataCombo_Change()
  sSQLQuery = "SELECT * FROM Item " + _
        "WHERE SubcategoryNumber IN (Select SubcategoryNumber FROM SubfoodCategory " + _
                        "WHERE SubcategoryName ='" + SubcategoryDataCombo.Text + "')"
  ItemAdodc.RecordSource = sSQLQuery
  ItemAdodc.Refresh
End Sub

Private Sub TableButtons_Click(Index As Integer)
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Or SpecialMarkFrame.Visible Then
  Else
    If tableSeatNumber(Index) <> 0 Then
      orderNumber = ""
      orderTableNumber = Index
      totalOrderItemNumber = 0
      readOrderItemNumber = 0
      mOrderForm.Caption = "Table Map"
      If TableButtons(Index).BackColor = TableAvailableColor Then
        str2 = "False"
        str3 = tableNumber(orderTableNumber)
        str4 = "charlie"
        UpdateTableavailableStatus
        ShowTable True
      End If
        If tableNo(orderTableNumber) < 10 Then
          tableNumberString = "Table No. 0" + CStr(tableNo(orderTableNumber))
        Else
```

```
            tableNumberString = "Table No. " + CStr(tableNo(orderTableNumber))
          End If
          ReadUncheckoutOrderInfo
          ShowStatusFrame True
        'End If
      End If
    End If
End Sub

Private Sub TableCoordinateConvertion()
  xRate = (TableMapFrame.Width - 8 * margin) / 3600
  yRate = (TableMapFrame.Height - 2 * margin) / 4020
  For i = 1 To totalTableNumber
    tableXCoordinate(i) = tableXCoordinate(i) * xRate
    tableYCoordinate(i) = tableYCoordinate(i) * yRate + 2 * margin
    tableWidth(i) = tableWidth(i) * yRate
    tableLength(i) = tableLength(i) * yRate
  Next i
End Sub


Private Sub Timer1_Timer()
    TimeLabel.Caption = " Time: " + CStr(Time) + " " + CStr(Date)
End Sub

Private Sub TogoButton_Click()
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Or SpecialMarkFrame.Visible Then
  Else
    'PaymentButton(3).Enabled = True
    tableNumberString = "To Go Order"
    orderTableNumber = counterTableNumber
    orderNumber = ""
    totalOrderItemNumber = 0
    alreadyOrderItemNumber = 0
    readOrderItemNumber = 0
    ShowTableMapFrame False
    ShowMenuFrame True
    ClearOrderItemLabels
    ShowOrderItemFrame True
  End If
End Sub

Private Sub ToGoListButton_Click()
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Or SpecialMarkFrame.Visible Then
  Else
    'PaymentButton(3).Enabled = True
    mOrderForm.Caption = "To Go List"
    tableNumberString = "To Go Order"
    ClearOrderItemLabels
    ReadToGoOrderInfo
    ShowOrderItemFrame True
  End If
End Sub

Private Sub UpdateAccountingInfo()
  sSQLQuery = "UPDATE Accounting SET EndDate ='" & CStr(Date) & "', EndTime ='" & CStr(endTime) & _
        "' WHERE AccountingID =1"
  oConn.Open dbEngine
  oConn.Execute sSQLQuery
  oConn.Close
End Sub

Private Sub UpdateOrderItem()
  If SPFrame.Visible Or KeyboardFrame.Visible Or PaymentFrame.Visible Or NumericalFrame.Visible Or _
    StatusFrame.Visible Or DiscountFrame.Visible Then
```

```
  ElseIf OrderItemNameLabel(orderItemIndex) <> "" Then
    If OrderFrameTitleLabel.Caption = "Order List" Then
      mOrderForm.Caption = "Order List"
      choiceItemName = OrderItemNameLabel(orderItemIndex)
      ShowStatusFrame True
    ElseIf OrderFrameTitleLabel.Caption = "Order Status" Then
      mOrderForm.Caption = "Order Status"
      choiceSeriesNumber = OrderItemNumberLabel(orderItemIndex)
      choiceItemName = OrderItemNameLabel(orderItemIndex)
      ShowStatusFrame True
    ElseIf TableNoLabel.Caption = "To Go Order" Then
      orderTableNumber = counterTableNumber
      totalOrderItemNumber = 0
      readOrderItemNumber = 0
      mOrderForm.Caption = "To Go List"
      choiceItemName = OrderItemDemandLabel(orderItemIndex)
      tableNumberString = choiceItemName
      orderNumber = OrderItemNumberLabel(orderItemIndex).Caption
      ReadOrderPaymentInfo
      ShowStatusFrame True
    End If
  End If
End Sub

Private Sub UpdateTableavailableStatus()
  sSQLQuery = "UPDATE Tables SET AvailableStatus =" & str2 & _
          ", WaiterNumber ='" & str4 & _
          "' WHERE TableNumber =" & str3 & _
          " AND AvailableStatus =" & Not (str2)
  oConn.Open dbEngine
  oConn.Execute sSQLQuery
  oConn.Close
End Sub

Private Sub WriteOrderDiscount()
  sSQLQuery = "UPDATE Orders SET Discount =" + CStr(discount) + " WHERE OrderNumber=" + orderNumber
  oConn.Open dbEngine
  oConn.Execute sSQLQuery
  oConn.Close
End Sub

Private Sub WholeDayButton_Click()
  If Not NumericalFrame.Visible Then
    totalSales = 0
    startTime = "00:00:00"
    endTime = "23:59:59"
    ReportLabel.Caption = CStr(Date) + " Daily Sales Report"
    ReadAllOrderItems
  End If
End Sub

Private Sub Yes_Click()
  End
End Sub
```

REFERENCES

[1] Li Qiu, Project Proposal: mOrder-Client Food Service, March 2002.

[2] Peter Wrights, Beginning Visual Basic 6: Wrox, Aug 2001.

[3] Deitel & Deitel Nieto, How to Program Visual Basic 6: Prentice Hall, 1999.

[4] Elmasri, Navathe, Fundamentals of Database System: Addison Wesley, 2000.

[5] Dot Matrix SP2000 Series Programmer's Manual.